A Turning Point For Atari?
Report From The Winter Consumer Electronics Show

COMPUTE!

$2.95
April
1986
Issue 71
Vol. 8, No.4
$3.75 Canada
02193
ISSN 0194-357X

The Leading Magazine Of Home, Educational, And Recreational Computing

New Technologies: The Converging Digital Universe
Why Everything From Computers & CD-ROMs To TVs & Satellites Are On A Collision Course

Adding Power To ST BASIC
Valuable Techniques For Atari Programmers

Apple Hi-Res Graphics
Mix, Flip, And Copy High-Resolution Screens

64 Screen Genie
A Magic Method For Character Graphics

AmigaDOS Batch Files
Take Control Of Your Amiga

Atari FontMaker
Design Your Own Custom Characters

(No I.D. required for half-elves.)

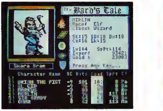
When the Going Gets Tough, the Bard Goes Drinking.

And the going is tough in Skara Brae town. The evil wizard Mangar has cast an eternal winter spell. Monsters control the streets and dungeons beneath. Good citizens fear for their lives. What's worse, there's only one tavern left that serves wine.

But the Bard knows no fear. With his trusty harp and a few rowdy minstrel songs he claims are magic, the Bard is ready to boogie. All he needs is a band of loyal followers: a light-fingered rogue to find secret doors, a couple of fighters to bash heads, a conjurer to create weird allies, a magician for magic armor.

Then it's off to combat, as soon as the Bard finishes one more verse.

Now what's a word that rhymes with "dead ogre?"



The Bard's Tale

MERLIN
Race: Elf
Class: Wizard

St:15 IQ:18 Dx:18
Cn:16 Lk:15

Lvl:4 SpPt:114
Exper: 93421
Gold : 23003

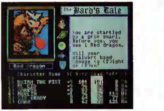
Press Any Key...

Skara Brae

Character Name AC Hits Cond SpPt Cl

BRIAN THE FIST
EL CID
MARKUS
MERLIN
SIR GRADY
OMAR

4 classes of magic user, including wizard and sorceror. 85 new magic spells in all.



The Bard's Tale

You are startled by a grim snarl. Before you, you see 1 Red dragon.

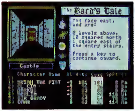
Will your stalwart band choose to (F)ight or (R)un?

Red dragon

Character Name AC Hits Cond SpPt Cl

BRIAN THE FIST
EL CID
MARKUS
MERLIN
SIR GRADY
OMAR

128 color monsters, many animated. All challenging.



The Bard's Tale

You face east, and are:

0 levels above, 10 squares north 1 square east of the entry stairs.

Press a key to continue onward.

Castle

Character Name AC Hits Cond SpPt Cl

BRIAN THE FIST
EL CID
MARKUS
MERLIN
SIR GRADY
OMAR

Full-color scrolling dungeons. 16 levels, each better than the one before. 3-D city, too.

Optional use of characters created in Wizardry® or Ultima III!

The Bard's Tale™

from

ELECTRONIC ARTS™

Specs: 100% machine language, 400K worth on 2 disks. 64K RAM minimum, enhanced for 128K. Now available for Apple II family for $44.95, C-64 & 128 for $39.95.
To order: Visit your retailer or call (800) 227-6703 (In Calif. call (800) 632-7979) for VISA or MasterCard orders. To purchase by mail, send check or money order to Electronic Arts, P.O. Box 306, Half Moon Bay, CA 94109. Add $3 for insured shipping and handling. Allow 1-4 weeks for delivery. The Bard's Tale is worth the wait.

Apple is a trademark of Apple Computer. Wizardry is a registered trademark of Sir-Tech Software, Inc. Ultima is a registered trademark of Richard Garriott. The Bard's Tale and Electronic Arts are trademarks of Electronic Arts. For a copy of our complete catalogue and direct order form, send $.50 and a stamped, self-addressed envelope to Electronic Arts, 2755 Campus Drive, San Mateo, CA 94403.

We don't care which computer you own. We'll help you get the most out of it.

CompuServe puts a world of information, communications, and entertainment at your fingertips.

CompuServe is the world's largest information service designed for the personal computer user and managed by the communications professionals who provide business information services to over one quarter of the FORTUNE 500 companies.

Subscribers get a wealth of useful, profitable or just plain interesting information like national news wires, home shopping and banking, travel and sophisticated financial data. Plus electronic mail, national bulletin boards, forums (special interest groups), and a multi-channel CB simulator.

You get games and entertainment, too. Board, parlor, sports, space and educational games. Trivia and the first online TV-style game show played for real prizes.

To buy a CompuServe Subscription Kit, see your nearest computer dealer. To receive our informative brochure or to order direct call or write:

CompuServe®

Consumer Information Service, P. O. Box 20212
5000 Arlington Centre Blvd., Columbus, OH 43220
800-848-8199 In Ohio Call 614-457-0802

An H&R Block Company

Free software from Electronic Arts.™

It's easy!

Buy any of these 12 smash hits from your participating dealer between February 1, 1986, and May 31, 1986. Then just mail in your proof of purchase, $5 for shipping and handling, and the official "Get One Free" coupon. So what are you waiting for?

In case of emergency...

If you can't find a participating dealer, you can order direct from us by telephone or mail. Just call toll-free 800-227-6703. In California call 800-632-7979. Have your Visa or MasterCard number ready. Just tell us what you want to buy, and what you want for free. It's simple. And there's a 14-day "satisfaction or your money back" guarantee on all products ordered direct.

In case your participating dealer runs out of "Get One Free" coupons, use this.

cut here

□ Yes! I want free software from Electronic Arts!

□ No thanks. I'm using my computer as a doorstop.

(Check one)

Apple
II, II+, //c, //e

- □ Dr J. & Larry Bird Go One-On-One
- □ Archon
- □ Financial Cookbook
- □ Music Construction Set
- □ Pinball Construction Set
- □ Seven Cities of Gold
- □ Archon II Adept
- □ Movie Maker

Commodore
64 & 128

- □ Dr J. & Larry Bird Go One-On-One
- □ Archon
- □ Financial Cookbook
- □ Music Construction Set
- □ Pinball Construction Set
- □ Seven Cities of Gold
- □ Archon II Adept
- □ Movie Maker
- □ M.U.L.E.
- □ Realm of Impossibility
- □ Mail Order Monsters
- □ Racing Destruction Set

Atari
400-1200 series

- □ Dr. J. & Larry Bird Go One-On-One
- □ Archon
- □ Financial Cookbook
- □ Music Construction Set
- □ Pinball Construction Set
- □ Seven Cities of Gold
- □ Archon II . Adept
- □ Movie Maker
- □ M.U.L.E.
- □ Realm of Impossibility

IBM
PCjr, PC, & comp.

- □ Dr J. & Larry Bird Go One-On-One
- □ Archon
- □ Financial Cookbook
- □ Music Construction Set
- □ Pinball Construction Set
- □ Seven Cities of Gold

Please send my free software to the following address. I have enclosed the required Proofs of Purchase and $5 (check or money order payable to Electronic Arts) to cover the cost of shipping and handling ($7 Canadian).

PLEASE PRINT

Name

Address

City State Zip

Phone ( )- -

Signature

Proof of Purchase Send us the following two items 1) the dated cash register tape or receipt showing the product you purchased, and 2) find the Command Summary or Reference Card inside the package you purchased Cut off the part number on the back lower-left-hand side of the card or the product name from the front of the card Send these two items and $5 for shipping and handling ($7 Canadian) along with the coupon to **Electronic Arts, "Buy 1, Get 1 Free," P. O. Box 7530, San Mateo, CA 94403.** This offer expires May 31, 1986 Open to residents of the U S A and Canada only Only the products listed are available for this promotion No substitutions will be accepted

(no dogs included . . .)

Let's face it. Buying home software can be a risky proposition. Good software costs a lot of money. Cheap software practically rolls over and barks. But look at this.The titles listed below are Electronic Arts' greatest hits. They've all scored on the "top twenty" best-seller charts. Between them they've won over 60 product quality awards—making them the most honored products in the home software industry. Best of all, there's no longer any reason to swallow hard when you think about treating yourself to quality software. Because now you can:

Buy 1, Get 1 Free.

DR J & LARRY BIRD
GO ONE-ON-ONE™
The best-selling computer sports simulation of all time
"Game of the Year" – Electronic Games
Apple // family, C-64 & 128
Atari, IBM PCjr, PC, and comp

PINBALL
CONSTRUCTION SET™
Build your own video pinball games
"Best Game Generator" – Infoworld
Apple // family, C-64 & 128, Atari,
IBM PCjr PC, and comp

MUSIC
CONSTRUCTION SET™
Composition for anyone who can point a joystick
Best Music Generator" – Infoworld
Apple // family, C-64 & 128, Atari,
IBM PCjr, PC, and comp

RACING
DESTRUCTION SET™
Land mines, oil slicks, animated spin-outs, collisions and crashes
" the ultimate racing game "
– Computer Entertainer
C-64 & 128

SEVEN CITIES OF GOLD™
Play Conquistador in this educational adventure of history and geography
'Best Role-Playing Adventure'
– Family Computing
Apple // family, C-64 & 128, Atari,
IBM PCjr, PC and comp

ARCHON™
Like chess with arcade battle action
"Game of the Year"
– Creative Computing
Apple // family, C-64 & 128, Atari,
IBM PCjr, PC, and comp

ARCHON II ADEPT™
Graduate school for Archon addicts
Even more strategy and magic
"Best Strategy/Arcade Game"
– Family Computing
Apple // family, C-64 & 128, Atari

MAIL ORDER MONSTERS™
New Like an 'Archon Construction Set"
"Best Construction Set"
– Family Computing
C-64 & 128

MOVIE MAKER™
Create your own high-quality animated movies
'A masterpiece"
– Creative Computing
Apple // family, C-64 & 128, Atari

REALM OF IMPOSSIBILITY™
Action and adventure in a world of Escher-like illusions Includes a unique 2-player cooperative mode
C-64 & 128, Atari

M U L E™
The best multi-player computer game of all time
"Best Strategy Game" – Infoworld
'Best Multi-Player Game"
– Electronic Games
C-64 & 128, Atari

FINANCIAL COOKBOOK™
Take control of your personal finances
Like a calculator, spreadsheet, and financial advisor combined
Apple // family, C-64 & 128,
Atari, IBM PCjr, PC, and comp

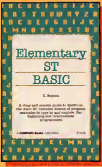
Remember—this offer is good only on these 12 products for the Apple // family, C-64 and 128, Atari, and IBM PC, PCjr, and compatible computers. No substitutions will be accepted For details, see the accompanying coupon

For a complete Electronic Arts Catalog send 50¢ and a stamped self addressed envelope to Electronic Arts Catalog 1820 Gateway Drive San Mateo CA 94404

IBM is a registered trademark of International Business Machines Inc Apple // is a regisrered trademark of Apple Computer Inc Commodore 64 & 128 are registered trademarks of Commodore Business Machines Inc Atari is a registered trademark of Atari Computer Corp Electronic Arts Dr J & Larry Bird Go One on One Archon Financial Cookbook Music Construction Set Pinball Construction Set Seven Cities of Gold Archon II – Adept Movie Maker M U L E Realm of Impossibility Mail Order Monsters and Racing Destruction Set are registered trademarks of Electronic Arts

ELECTRONIC ARTS™

Elementary Amiga BASIC
C. Regena
0-87455-041-6, $14.95
Disk $15.95



Elementary ST BASIC
C. Regena
0-87455-034-3, $14.95
Disk $15.95

BASIC programming at its best!

Two new programming guides from COMPUTE! Books.

Written by the author of the bestselling *Programmer's Reference Guide to the TI-99/4A*, these books introduce you to the new and powerful BASIC on the Amiga and Atari ST personal computers. The computers' impressive graphics, animation, and sound can be unlocked with the right commands, and BASIC is the place to start. Regena shows you how—in the clear, concise language that's made her such a popular writer.

Complete descriptions of the Amiga's and ST's BASIC commands, syntax, and organization take you from novice to full-power programming. Sample programs and subroutines, all ready to type in, are included. Plus, both books offer you working software while showing you how to write your own programs. *A disk is also available for each book which includes all the programs from the book in an easy, ready-to-load format.*

SPECIAL COMBINATION OFFER
Order the book and disk together for only $29.95!

You'll find these new programming guides and many more useful, entertaining COMPUTE! books at your local computer and book stores. Or you can order directly from COMPUTE! Books.

For the fastest service, call toll free 1-800-346-6767 (in NY 212-887-8525). Or mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please add shipping and handling charges to all orders: In U.S., $2.00 per book, disk, or combination package; $5.00 per item airmail.

I want the best in BASIC programming guides. Please send me:

______ ***Elementary ST BASIC,*** (034-3), $14 95 each ______
______ ***Elementary ST BASIC Disk***, $15 95 each ______
______ ***Elementary ST BASIC*** disk and book combination, $29 95 ______
______ ***Elementary Amiga BASIC,*** (041-6), $14.95 each ______
______ ***Elementary Amiga BASIC Disk,*** $15.95 each ______
______ ***Elementary Amiga BASIC*** disk and book combination, (057-2), $29 95 ______

Subtotal ______
NC residents add 4.5% sales tax ______
Shipping charges ______
($2 00 per item in U.S and surface mail.
$5 00 per item airmail)
Total amount enclosed ______

ALL ORDERS
MUST BE
PREPAID IN
U S FUNDS

□ Payment enclosed (check or money order)
□ Charge □ MasterCard □ Visa □ American Express
Account No. ______ Exp. Date ______
(Required)
Name ______
Address ______
City ______
State ______ Zip ______
Allow 4–6 weeks for delivery
36412112

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines Inc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE COMPUTE!'s Gazette COMPUTE's Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England and in Canada from McGraw-Hill, Ryerson Ltd., 330 Progress Ave., Scarborough, Ontario, Canada M1P 2Z5.

COMPUTE!

APRIL 1986
VOLUME 8
NUMBER 4
ISSUE 71

FEATURES

REVIEWS

COLUMNS AND DEPARTMENTS

THE JOURNAL

NOTE: See page 125 before typing in programs.

AP Apple, **Mac** Macintosh, **AT** Atari, **ST,** Atari ST, **V** VIC-20, **64** Commodore 64, **+4** Commodore Plus/4, **16** Commodore 16, **128** Commodore 128, **P** PET/CBM, **TI** Texas Instruments, **PC** IBM PC, **PCjr** IBM PCjr, **AM** Amiga *General interest.

TOLL FREE Subscription Order Line
800-247-5470 (In IA 800-532-1272)

COMPUTE! Publications, Inc. abc
Part of ABC Consumer Magazines, Inc.
One of the ABC Publishing Companies
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019

COMPUTE! The Journal for Progressive Computing (USPS: 537250) is published monthly by COMPUTE! Publications, Inc., 825 7th Ave., New York, NY 10019 USA. Phone. (212) 265-8360. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions 12 issues, $24. POSTMASTER: Send address changes to. **COMPUTE!** Magazine, P.O. Box 10955, Des Moines, IA 50950. Second class postage paid at Greensboro, NC 27403 and additional mailing offices. Entire contents copyright ©1986 by COMPUTE! Publications, Inc. All rights reserved, ISSN 0194-357X.

Editor's Notes

COMPUTE! *Editor Tom Halfhill contributes an editorial this month.*
—Robert Lock, Editor in Chief

The latest-generation personal computers present the best evidence to date that microcomputers are evolving into desktop mainframes. Megabytes of memory, hard disk drives, high-speed processors, and multitasking operating systems are no longer limited to the monster machines locked away in the data processing departments of governments and big corporations. Now you can get these features in a personal computer that costs less than $2,000 and fits comfortably on a desktop.

But why would you want to? After all, many people are questioning why anyone needs *any* kind of computer in their home. Are the new machines just a more blatant example of technological overkill?

Practically everyone who's ever used a computer understands the value of more memory and mass storage, high-speed processing, and faster input/output. All those things translate into more horsepower, and if the price is right, we'll welcome more horsepower. But one feature that some people are regarding with skepticism is multitasking—the ability to run more than one program at a time. Is it really practical to run a spreadsheet and a word processor simultaneously? Even if the computer can do two things at once, the user probably can't.

This criticism overlooks several advantages of multitasking: its convenience, the way it shifts busy work away from the user and onto the computer, its implications for software design, and its future applications in tomorrow's homes.

It's hard to appreciate the sheer convenience of multitasking until you've experienced it. Even if you aren't actively *using* two or more programs at once, you can keep them loaded in memory, available at the press of a key or click of a mouse button. For instance, you can type a letter with a word processor, switch to a terminal program to upload it to an electronic mail service, then switch to BASIC to finish a program you've been writing. On most home computers, that would require rebooting the machine several times, swapping disks, running different programs, and waiting.

Multitasking can also spare you some drudgery by letting the computer do the tedious jobs. If you log onto a commercial information service to check stock quotations every evening, you can set up the computer to do this for you automatically—even while you're using the machine for something else in the meantime. Multitasking is something that's hard to do without once you've had it.

Multitasking also lets you create your own integrated software packages. You can buy whatever word processor, graphics program, spreadsheet, and terminal program you want and load them all into memory at once. If the computer supports a standardized file transfer protocol—as do the Macintosh and Amiga—you can cut and paste pictures or spreadsheet tables into documents created with the word processor and so forth, even if the programs were made by different software companies.

Finally, there are exciting possibilities for multitasking in the future. Remember that microcomputers are following the paths established by mainframes; it's a small step from multitasking to multiuser processing. If a computer can run several programs at once in windows on a single screen, why not turn those windows into separate screens and put them in different rooms? We know from our mail and readership surveys that many of you are already multicomputer households. Mom and Dad have a computer in the study, and the kids have one or two in the family room or bedroom. Someday you'll be able to buy a single personal computer with enough brute force to drive several terminals throughout your home. Each terminal will be as powerful and seemingly as independent as today's personal computers, yet the system will be economical because you'll all share the same printer, modem, hard disk drive, and CD-ROM player.

The main disadvantages of multitasking—the amounts of memory and processing time it can gobble up—are temporary annoyances. Memory chips are getting cheaper as fast as microprocessors are growing more powerful. Atari recently introduced the first 1024K computer for under $1,000, just five years after an 8K Atari 800 retailed for $1,000. And Motorola recently announced a 20 megahertz version of its 68020 microprocessor, referred to as the "mainframe on a chip."

It seems that the only real problem to be overcome is the incredible complexity of writing and debugging a true multitasking operating system. Ask an IBM owner about all the popular *Sidekick*-type, co-resident programs that compete for the attention of DOS interrupts and the keyboard. Or ask an Amiga owner about the weird things that can happen when the computer tries to do too much at once. (In fact, one of the strangest things we've seen on the Amiga is something that can be described as a "half crash." On practically any other computer, a system crash is a system crash—the machine locks up and you have no choice but to reboot. On the Amiga, we've managed to crash *part of the computer* while the other part struggles valiantly onward. You end up rebooting anyway just to play safe, but it's an interesting demonstration of multitasking.)

Essentially, multitasking gives you the near-equivalent of several computers in a single box. And if the box is priced right and meets your other requirements, why walk when you can run?

Tom R. Halfhill, Editor

IF YOU CAN FIND A BETTER PROGRAM WE'LL BUY IT FOR YOU!*

WORD WRITER with Spell Checker

Now with 85,000 word Spelling Checker

- An efficient, 80-column professional word processing system which includes a spelling checker and built-in calculator.
- Contains all the features you'll need for everyday word processing, plus most of the sophisticated features found in more expensive programs: document chaining, form letter printout, page separations, horizontal and vertical scrolling, and more.

With Timeworks you get more power for your dollar

You can use each program alone. Or interface this trio – one at a time if you like – into a completely integrated productivity system that delivers all the power and features most of you will ever need . . . at a cost that lets you enjoy their use.

Look for these and other Timeworks programs at your favorite dealer. Or contact Timeworks, 444 Lake Cook Road, Deerfield, IL 60015. Phone: (312) 948-9200.

DATA MANAGER 2

Faster, more efficient, more versatile

- A complete general information storage and retrieval system with report-writing, graphics, statistics, and label-making capabilities.
- Exclusive X-SEARCH, X-SORT, and X-CHART features allow you to cross-search any category of information; sort items alphabetically, numerically, or by date; break down statistical information into categories; and graphically view your results.

With Timeworks you get more than software

You Get Our Customer Technical Support Team

At the other end of our toll-free hotline, you'll find our full-time Customer Technical Support Team. Free of charge to all registered users.

You Get Our Liberal Trade-Up Policy

You'll find the details inside each package.

SWIFTCALC with SIDEWAYS

New easy-to-use spreadsheet for home and small business use

- The SIDEWAYS option lets you print all your columns on one, continuous sheet . . . sideways.
- 250 rows and up to 250 columns (128K version) provide up to 62,500 cells (locations on the spreadsheet) in which to place information.
- Performs mathematical functions up to 17 digits. Allows the use of minimum and maximum values, averages, sums, integers, absolute values and exponential notation.

***With Timeworks you get our Money Back Guarantee**

If you can find anything that works better for you, simply send us your Timeworks program, your paid receipt, and the name of the program you want, along with your check or credit card number for any retail price difference. If it's available, we'll buy it for you.**

**For Apple,
Commodore 128 (128K)
& Commodore 64
Computers**

TIMEWORKS®

More power for your dollar.

Other Timeworks Programs:

- The Evelyn Wood Dynamic Reader
- Sylvia Porter's Personal Finance Series
- Swiftax ■ Cave of the Word Wizard
- Business Systems ■ Wall Street
- The Electronic Checkbook
- The Money Manager

These programs interface with each other

Now available for Commodore 128

** Offer valid for 90 days from date of purchase

*** Registered trademarks of Apple Computer, Inc., International Business Machines Corporation, and Commodore Computer Systems

© 1985 Timeworks Inc All rights reserved

Publisher	James A. Casella
Founder/Editor in Chief	Robert C. Lock
Senior Editor	Richard Mansfield
Managing Editor	Kathleen Martinek
Executive Editor	Selby Bateman
Editor	Tom R. Halfhill
Assistant Editor	Philip Nelson
Production Director	Tony Roberts
Production Editor	Gail Cowper
Editor, COMPUTE!'s GAZETTE	Lance Elko
Technical Editor	Ottis R. Cowper
Assistant Technical Editors	John Krause, George Miller
Program Editor	Charles Brannon
Assistant Editor, COMPUTE!'s GAZETTE	Todd Heimarck
Assistant Features Editor	Kathy Yakal
Programming Supervisor	Patrick Parrish
Editorial Programmers	Tim Victor, Kevin Mykytyn
Research/Copy Editor	Joan Rouleau
Copy Editor	Ann Davies
Submissions Reviewer	Mark Tuttle
Programming Assistants	David Florance, David Hensley
Executive Assistant	Debi Nash
Administrative Assistants	Julia Fleming, Iris Brooks, Mary Hunt, Sybil Agee
Associate Editors	Jim Butterfield Toronto, Canada Harvey Herman Greensboro, NC Fred D'Ignazio Roanoke, VA David Thornburg Los Altos, CA
Contributing Editor	Bill Wilkinson
COMPUTE!'s Book Division Editor	Stephen Levy
Assistant Editor	Gregg Keizer
Director, Book Sales & Marketing	Steve Voyatzis
Production Manager	Irma Swain
Art & Design Director	Janice R. Fary
Assistant Editor, Art & Design	Lee Noel
Mechanical Art Supervisor	De Potter
Artists	Debbie Bray, Dabney Ketrow
Typesetting	Terry Cash, Carole Dunton
Illustrator	Harry Blair
Director of Advertising Sales	Ken Woodard
Production Coordinator	Kathleen Hanlon
Promotion Assistant	Caroline Dark
Customer Service Manager	Diane Longo
Dealer Sales Supervisor	Orchid Tamayo
Individual Order Supervisor	Judy Taylor
Receptionist	Anita Armfield
Warehouse Manager	John Williams
Data Processing Manager	Leon Stokes

James A. Casella, President
R. Steven Vetter, Vice President, Finance and Planning

COMPUTE! Publications, Inc. publishes

COMPUTE!
The Journal For Progressive Computing

COMPUTE!'s GAZETTE
For VIC-20 and Commodore 64 Personal Computers

COMPUTE! Books

COMPUTE!'s GAZETTE DISK

COMPUTE!'s
Apple Applications Special

Editorial offices:	324 West Wendover Avenue Suite 200 Greensboro, NC 27408 USA
Corporate offices	825 7th Avenue New York, NY 10019 212-265-8360
Customer Service	800-346-6767 (In NY 212-887-8525)

Coming In Future Issues

Two New Columns! For Atari ST And Amiga

Hickory Dickory Dock: Teaching Kids To Tell Time For Commodore, Atari, ST, Amiga, Apple, PC/PCjr

Commodore Auto-Boot Maker

Custom Characters For Atari *SpeedScript*

Add System Power To Atari ST BASIC, Part 2

Amiga Puzzle: An Amiga BASIC Application

Windows On The Commodore 128

Member
Audit Bureau of Circulations

MPA
Magazine Publishers Association

Subscription Orders

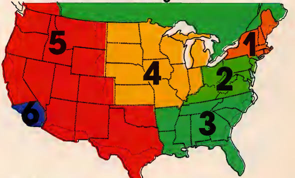
COMPUTE!
P.O. Box 10954
Des Moines, IA 50340

TOLL FREE Subscription Order Line
800-247-5470
In IA 800-532-1272

COMPUTE! Subscription Rates (12 Issue Year):

US	(one yr.)	$24
	(two yrs.)	$45
	(three yrs.)	$65
Canada and Foreign Surface Mail		$30
Foreign Air Delivery		$65

Advertising Sales



1. New England
John Saval
Eastern Advertising Manager
212-315-1665

2. Mid Atlantic
John Saval
Eastern Advertising Manager
212-315-1665

3. Southeast & Foreign
Harry Blair
919-275-9809

4. Midwest
Gordon Benson
312-362-1821

5. Northwest/ Mountain/Texas
Phoebe Thompson
Dani Nunes
408-354-5553

6. Southwest
Ed Winchell
213-378-8361

Director of Advertising Sales
Ken Woodard

COMPUTE! Home Office 212-887-8460

Address all advertising materials to:
Kathleen Hanlon
Advertising Production Coordinator
COMPUTE! Magazine
324 West Wendover Avenue
Suite 200
Greensboro, NC 27408

The COMPUTE! subscriber list is made available to carefully screened organizations with a product or service which may be of interest to our readers. If you prefer not to receive such mailings, please send an exact copy of your subscription label to: COMPUTE! P.O. Box 10955, Des Moines, IA 50950. Include a note indicating your preference to receive only your subscription.

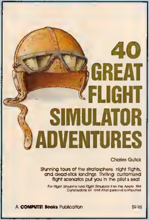
Authors of manuscripts warrant that all materials submitted to COMPUTE! are original materials with full ownership rights resident in said authors. By submitting articles to COMPUTE!, authors acknowledge that such materials, upon acceptance for publication, become the exclusive property of COMPUTE! Publications, Inc. No portion of this magazine may be reproduced in any form without written permission from the publisher. Entire contents copyright © 1986 COMPUTE! Publications, Inc. Rights to programs developed and submitted by authors are explained in our author contract. Unsolicited materials not accepted for publication in COMPUTE! will be returned if author provides a self-addressed, stamped envelope. Programs (on tape or disk) must accompany each submission. Printed listings are optional, but helpful. Articles should be furnished as typed copy (upper- and lowercase, please) with double spacing. Each page of your article should bear the title of the article, date and name of the author. COMPUTE! assumes no liability for errors in articles or advertisements. Opinions expressed by authors are not necessarily those of COMPUTE!

PET, CBM, VIC-20 and Commodore 64 are trademarks of Commodore Business Machines, Inc. and/or Commodore Electronics Limited
Apple is a trademark of Apple Computer Company
IBM PC and PCjr are trademarks of International Business Machines, Inc.
ATARI is a trademark of Atari, Inc.
TI-99/4A is a trademark of Texas Instruments, Inc.
Radio Shack Color Computer is a trademark of Tandy, Inc.

Discover the thrill of *flying!*

with COMPUTE! Books'
40 Great Flight Simulator Adventures

- •Maneuver around the towers of the World Trade Center.
- •Sightsee the Hudson River.
- •Practice night flying and aerobatics.



40 GREAT FLIGHT SIMULATOR ADVENTURES

Charles Gulick

Stunning tours of the stratosphere, night flights, and dead-stick landings. Thrilling, customized flight scenarios put you in the pilot's seat.

For Flight Simulator and Flight Simulator II on the Apple, IBM, Commodore 64, and Atari personal computers

A **COMPUTE! Books** Publication $9.95

40 Great Flight Simulator Adventures

Charles Gulick

Forty exciting, customized flight simulator scenarios put you in the pilot's seat as you fly over bridges, around skyscrapers, and land at mysterious airports. *Flight Simulator* (IBM PC) and *Flight Simulator II* (Apple II, Commodore 64, Atari) are two of the most popular games/simulations for personal computers. With this book, you can experience flight adventures from the moment you load the program. Parameters set up each flight and a running commentary describes what you'll see (and where to look to see it). Ranging from the simple and straightforward to the advanced and even mystical, these 40 flights will open a new dimension to an already outstanding program.

$9.95 ISBN 0-87455-022-X

Note: *Flight Simulator* from Microsoft Corporation and *Flight Simulator II* from Sublogic Corp. are required in order to use this book. The book is designed to enhance the programs.

Please send me ________ copies of *40 Great Flight Simulator Adventures* at $9.95 each.
(ISBN No. 0-87455-022-X)

All orders must be prepaid in U.S. funds.

Subtotal ________
NC residents add 4.5% tax ________
$2.00 shipping and handling charge per book. ________
Total amount enclosed ________

To order this exciting adventure guide, mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, Ny 10150. Or call toll free 1-800-346-6767 (in NY 212-887-8525.)

□ Payment enclosed (check or money order)

□ Charge □ Visa □ MasterCard □ American Express

Name ________________________________

Address ________________________________

City ____________________ State ____________ Zip ________

Please allow 4–6 weeks for delivery. 36402211

COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England.

Readers Feedback

The Editors and Readers of COMPUTE!

If you have any questions, comments, or suggestions you would like to see addressed in this column, write to "Readers' Feedback," COMPUTE!, P.O. Box S406, Greensboro, NC 27403. Due to the volume of mail we receive, we regret that we cannot provide personal answers to technical questions.

Relocating Machine Language

I would like to combine two Commodore machine language programs that both reside at location 491S2 ($C000). I know that BASIC lets you relocate programs quite easily, just by moving the bottom-of-BASIC pointer upward. How is this done with ML programs?

Richard Sands

Machine language programs written for a 6S02-based computer are usually quite difficult to relocate. For instance, say that you have an ML program at $C000 which starts with these instructions:

```
LDA  $C030,X
JSR  $C200
JMP  $C400
```

None of these instructions can be relocated unless you change the address contained in the instruction itself. The first (LDA $C030,X) retrieves one byte of data from a table beginning at location $C030 (note that the data lies within the program code). The JSR instruction works like GOSUB in BASIC, so JSR $C200 goes to a subroutine located at $C200 and then returns. JMP works like GOTO in BASIC: JMP $C400 sends the computer straight to the segment of code located at $C400. Now say that you move the entire program down to location $8000. The instruction JSR $C200 still sends the computer to $C200, but that address isn't within the program any more. To make the code work correctly at $8000, you'd have to change these three instructions to the following:

```
LDA  $8030,X
JSR  $8200
JMP  $8400
```

That's not particularly difficult, and some machine language monitors even have a special command to make such adjustments automatically. However, you must be careful not to change addresses that refer to locations outside *the program:*

```
JSR $FFD2
```

This instruction calls the standard Commodore print-a-character routine, located in the computer's ROM. If you mistakenly adjust this address along with all the internal address references, the result may be disastrous. Now let's look at a more difficult case:

```
LDA ($FB),Y
```

This instruction uses the powerful and very common indirect Y *addressing mode, which refers indirectly to an address held in two successive zero page addresses (locations $FB–$FC in this case). There's no way to tell by looking at this instruction alone whether it refers to an area inside the program (and hence requires adjustment) or something external to the program code (in which case adjustment may be a mistake). You'll have to disassemble the program in its entirety, looking for other instructions that affect the contents of locations $FB–FC, either directly or indirectly. If this instruction is part of a general-purpose subroutine, you may find that it's called by many different parts of the program. Since free zero-page space is limited, you may also find that other subroutines re-use locations $FB–FC for an entirely different purpose. And while it's obvious that an instruction like STA $FB affects the contents of $FB, what about ROR $03,X or STA ($B0),Y? Those instructions might just as easily change the address held in $FB–FC.*

Once you've sorted out all the indirect addressing, you'll need to check for self-modifying *routines—code that changes its own instructions while it runs. When that's done, you'll have to interpret all the program's data and variable areas. For instance, say that you find the following hexadecimal values in a memory dump of the program code:*

```
93 05 20 C4 54 0D 41 43
```

These bytes could be virtually anything—sprite shape data, characters for a printed message, part of an internal dispatch table, preset values for a bunch of unrelated variables, or even garbage that will be replaced with something meaningful when the program runs. While some programmers locate data areas at the end of the program, others sprinkle data and variables freely throughout the code. Until you find out exactly what purpose these bytes serve, there's no way to tell whether they need adjustment. This problem, more than any other, makes it impossible to write an "automatic ML relocator" that works correctly in every case. The relocator would need to have as much intelligence as a knowledgeable ML programmer who thoroughly understands the subject program.

These problems generally don't apply to 68000-based computers like the Amiga, Atari S20ST, and Macintosh. Since the computer normally decides for itself where to load the ML code, most 68000 ML programs must *be relocatable. That's no great hardship for programmers, since the 68000 instruction set includes many relocatable instructions.*

128 Atari Colors

Here is a machine language program that allows your Atari computer to display 128 colors at the same time. The program displays a different color on each horizontal display line.

```
10 FOR I=0 TO 21:READ A:P
   OKE 1536+I,A:NEXT I
20 POKE 752,1:PRINT CHR$(
   125):A=USR(1536)
30 DATA 173,11,212,201,32
   ,208,249,141,10,212,14
   2,24,208,232,232,208,2
   46,142
40 DATA 24,208,240,232
```

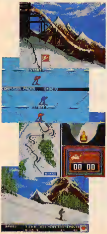
David Boyer

Thank you for the example.

Using Preview-80 With 64 *SpeedCalc*

I own a Commodore 64 and look forward to getting new programs from COMPUTE! each month, especially utilities. After typing in *SpeedCalc* (COMPUTE!, January 198S), I was pleased to find that the "Preview-80" program (COMPUTE!'s GAZETTE, November 198S) works just as well with *SpeedCalc* as it does with *SpeedScript*. This lets you preview a *SpeedCalc* file in 80-column format on the screen before printing it out. The procedure for using Preview-80 is the same as usual. First, load Preview-80 with LOAD"PREVIEW80",8,1. Then type NEW, and load *SpeedCalc* as you would normally. Instead of typing

WINTER GAMES.™ THE QUEST FOR THE GOLD CONTINUES...



You've captured the gold in Summer Games® and Summer Games II™. Now it's on to the *Winter Games*! And what an incredible setting—a completely realistic winter wonderland featuring *seven* action-packed events.

At the Ski Jump you control your form in mid-air, knees straight, leaning forward. Hot Dog Aerials challenges your courage *and* your sense of humor. In Figure Skating you leap into Double and Triple Lutz jumps—wow the crowd with a perfect Camel into a Sit Spin. It's timing and style that counts. Free Skating lets you choreograph your own routines. In Speed Skating it's you against a fellow speed demon—the fastest human beings on level earth! And the Bobsled—still faster as you fly around hairpin turns, leaning hard to stay in the tube. Finally the Biathlon, the ultimate challenge to your endurance in cross-country skiing and marksmanship.

All of this fun and excitement is easy to learn and play. You control the action with the joystick, animating your player for style and rhythm. You choose the country you want to represent. Listen to its national anthem. Then it's practice, training and learning a winning strategy for each event. Now the *Opening Ceremony* and the competition begins—against your friends or the computer. Will you be the one who takes the gold at the *Awards Ceremony*? Will your name be etched amongst the *World Record* holders?

The quest for the gold continues... And it's all here—the strategy, the challenge, the competition, and *pageantry* of Winter Games!

	APPLE	MAC	C64/128
Winter Games	✓	✓	✓

EPYX® COMPUTER SOFTWARE
1043 Kiel Ct., Sunnyvale, CA 94089

Strategy Games for the Action-Game Player®

RUN to start *SpeedCalc*, type SYS52000 and press RETURN. *SpeedCalc* will become active as usual. But when you press SHIFT-CTRL-P for printed output, and then press S for output to the screen, Preview-80 takes over. All of the Preview-80 options are available; to exit the Preview-80 window, press RUN/STOP twice.

Bob Starr

Thanks for the tip.

Moving AmigaDOS Commands To RAM

Regarding your article "Introduction to Amiga*DOS*" (COMPUTE!, January 1985), I feel that it's inconvenient to have all of the Amiga*DOS* commands—especially often-used commands like DIR—stored on disk rather than in memory. Is it possible to load all or part of DOS into RAM? If so, how much memory does it take up? Can you write a batch file to make this part of the boot sequence? Will Amiga*DOS* become RAM-resident in the future?

Barry Silverstein

Every AmigaDOS command is disk-resident, and you're not likely to see any change in the near future. This can be inconvenient at times since, for each separate AmigaDOS command, the computer has to access the same Workbench disk that was present when you booted the system. If you have only one disk drive, this scheme creates delays and requires extra disk-swapping. Fortunately, there's a simple remedy. If you create a RAM disk, you can then COPY any or all of the AmigaDOS commands from floppy disk to RAM disk; the amount of memory consumed depends on how many commands you copy. Once that's done, an ASSIGN command tells the system to use the RAM-resident commands.

The most convenient way to move AmigaDOS commands into RAM is by editing the startup-sequence *file, which is similar to an AUTOEXEC.BAT file in PC/DOS and MS/DOS systems. When you insert a disk in response to the Amiga's Workbench disk prompt, the computer looks in the S subdirectory of the currently mounted disk for a file named startup-sequence. If this file is present, the computer executes the AmigaDOS commands that it contains. Since startup-sequence is an ordinary ASCII text file, it's easy to modify with a word processor or any text editor that handles ASCII files. (Before editing this file, make sure that you have at least one copy of the Workbench disk* in *addition to the one that came with your computer.) If you edit this file with* Textcraft *or some other word processor, you must resave it in the form of plain ASCII text, without special formatting characters or control codes.*

AmigaDOS includes two text editors of its own. The easiest one to use is called ED. Type this line at the CLI prompt, then press RETURN:

```
ed "s/startup-sequence"
```

This command activates ED and loads startup-sequence into the editor. An unmodified startup-sequence file looks like this:

```
ECHO "Workbench disk. Version 1.1"
ECHO " "
ECHO "Use Preferences tool to set date."
ECHO " "
LoadWb
endcli > nil:
```

You'll probably recognize the messages that appear on the screen when you boot up with that disk. The LoadWb command loads and activates the Workbench, and endcli terminates the AmigaDOS command sequence, returning you to the Workbench screen. We'll use ED to add some new command lines between LoadWb and endcli. ED is a very simple text editor: Use the cursor keys to move around in the file, and the BACKSPACE key to delete characters. Everything that you type is inserted at the current cursor position (you can use uppercase if you like, but lowercase works just as well and is easier to type).

While you could copy the entire command directory (named C) into the RAM-disk, that wastes a lot of RAM since some AmigaDOS commands are used only rarely. To save memory, we'll copy only the most commonly used commands. Place the cursor on top of the E in endcli and enter these lines, pressing RETURN at the end of each line:

```
echo "Copying AmigaDOS commands to
      RAM disk..."
copy c/copy ram:c/copy
assign x: ram:c/copy
assign d: ram:c
cd sys:c
x: assign d:
x: cd d:
x: copy d:
x: delete d:
x: dir d:
x: diskcopy d:
x: echo d:
x: ed d:
x: endcli d:
x: info d:
x: list d:
x: makedir d:
x: newcli d:
x: rename d:
x: run d:
x: type d:
cd sys:
assign c: ram:c
assign d: c:delete
```

Remember, this set of commands goes between the LoadWb and endcli lines in the normal startup-sequence file. If you change your mind and don't want to modify the file, press ESC-Q followed by RETURN; ED returns you to the CLI without changing anything. To save the modified file to disk, press ESC-X followed by RETURN. After the file is resaved, ED returns you to the CLI prompt. To test the new startup-sequence file, reboot the computer by pressing CTRL-Left Amiga-Right Amiga. It takes about a minute to copy the commands shown above. Once the process is finished, all of the copied commands are instantly available in RAM (if this doesn't work, reload startup-sequence into ED and check for typing mistakes).

The first command line following ECHO copies the COPY command itself into RAM so the computer can copy subsequent commands without accessing the disk each time. The next three lines simplify your typing job: The first ASSIGN command tells the computer to substitute the characters ram:c/copy *wherever it sees the characters* x:. *The second ASSIGN creates another short alias (d:) which stands for the pathname* ram:c. *The CD command changes the current directory to SYS:C so you won't need to specify a subdirectory for every file you want to move. These three shortcuts let you abbreviate all of the remaining COPY commands (the command* x: endcli d: *becomes the equivalent of* ram:c/copy sys:c/endcli ram:c, *and so on).*

Thus, each line beginning with x: causes the computer to copy a single AmigaDOS command to the RAM disk. Of course, you can delete commands from this list, or add others if desired. The command ASSIGN C: RAM:C tells the computer to use the C directory in the RAM disk as its command directory. From this point on, the Amiga searches the RAM disk when you tell it to execute an AmigaDOS command. The final ASSIGN command isn't really necessary, but shows how to create a shorthand name for an often-used command. In this case, we're creating d: *as a synonym for DELETE. Once this is done, you can delete the file TEST by typing either DELETE TEST or D:TEST. This can be done for any command, using whatever shorthand you like. The command sequence shown here is adapted from an example in* COMPUTE!'s AmigaDOS Reference Guide, *which explains this and many other AmigaDOS topics in detail.*

HELP For Atari XL And XE

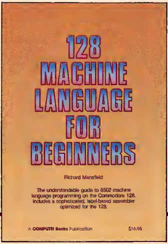
I have an Atari 800XL and would like to know how to read the HELP key.

R.E. Brock

The status of the HELP key can be determined by PEEKing location 732 on the Atari XL and XE computers. If the HELP key alone is pressed, this location returns a value of 17; when SHIFT and HELP are pressed simultaneously, it contains 81.

TAP THE POWER of the Commodore 128

By the author of *Machine Language for Beginners* and *Second Book of Machine Language*


128
MACHINE
LANGUAGE
FOR
BEGINNERS
Richard Mansfield
The understandable guide to 6502 machine language programming on the Commodore 128. Includes a sophisticated, label-based assembler optimized for the 128.
A COMPUTE! Books Publication
$16.95

128 Machine Language for Beginners
Richard Mansfield

One of the bestselling computer books ever has now been completely revised for the Commodore 128. Most commercial software is written in machine language because it's far faster and more versatile than BASIC. This new edition of *Machine Language for Beginners* is a step-by-step introduction to 8502 machine language programming on Commodore's 128 computer.

The book includes everything you need to learn to effectively program the 128: numerous programming examples, memory management tutorials; a complete description of the many Kernal routines and other new 128 features; numerous hints and programming techniques; and a dictionary of all major BASIC commands and their machine language equivalents. It also includes a high-speed, professional-quality, label-based assembler, optimized to take advantage of the speed and extra memory of the 128.
0-87455-033-5
$16.95

Like tha othar top-quality books from COMPUTEI, *128 Machine Language for Beginners* brings you ready-to-use information in a clear, lively style that makes learning easy and enjoyable, whether you are a beginner or an advanced computer user.

An optional disk is also available which includes tha assembler and example programs in the book. The *128 LADS Disk* is fully tested and ready to load on tha Commodore 128. It costs only $12.95 and saves you hours of typing tima.

Order your copy of *128 Machine Language for Beginners* and the *LADS Disk* today. Call toll free 1-800-346-6767 (in NY 1-212-887-8525) or mail your payment (plus $2.00 shipping par book or disk) to COMPUTEI Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books ere available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, Englend and in Canada from Holt, Rinehert, & Winston, 55 Horner Avenue, Toronto, ON M8Z 4X6.

COMPUTER MAIL ORDER

THE CMO ADVANTAGE

- THE BEST PRICES!
- Next day shipping on all in stock items
- Free easy access order inquiry
- Orders from outside Pennsylvania save state sales tax
- Free technicial support with our factory trained technical staff
- There is no limit and no deposit on C.O.D orders
- There's no extra charge for using your MasterCard or Visa. Your card is not charged until we ship
- No waiting period for cashiers checks.
- We accept purchase orders from qualified corporations. Subject to approval.
- Educational discounts available to qualified institutions
- FREE CATALOG MEMBERSHIP

ORDER LINE

CALL TOLL-FREE
1-800-233-8950
Educational Institutions
Call Toll-Free
1-800-221-4283
CUSTOMER SERVICE
& TECH SUPPORT
1-717-327-1450
Dept. A204

MAILING ADDRESS

Computer Mail Order
Dept. A204
477 East Third Street
Williamsport, PA 17701

MEMBER DIRECT MARKETING ASSOCIATION

CREDIT CARDS

VISA MasterCard

SHIPPING

Add 3%, minimum $7.00 shipping and handling on all orders. Larger shipments may require additional charges

All items subject to availability and price change.

Returned shipments may be subject to a restocking fee.

CANADIAN ORDERS

1-800-268-3974
Ontario/Quebec

1-800-268-4559
Other Provinces

1-416-828-0866
In Toronto

TELEX: 06-218960

2505 Dunwin Drive,
Mississauga, Ontario
Canada L5L1T1

All prices shown are tor U.S.A orders.
Call The Canadian Office for Canadian prices

HOME COMPUTERS

APPLE

APPLE IIe	CALL
APPLE IIc	CALL
MacINTOSH	CALL
IIc LCD Display	CALL

ATARI

65XE (64K)	**CALL**
130XE (128K)	**CALL**
520ST (512K)	**CALL**
800XL 64K	CALL
1010 Recorder	$49.99
1050 Disk Drive	CALL
1020 Printer	$29.99
1027 Letter Quality Printer	$129.00
1030 Direct Connect Modem	$59.99

COMMODORE

C128 Computer	**$269.00**
C1571 (Disk Drive for C128)	**$249.00**
C1902 (RGB 13" Monitor for C128	**CALL**
C1670 (Modem for C128)	**$179.00**
C64 Computer	CALL
C1541 Disk Drive	$199.00
C1530 Datasette	$39.99
C1802 Color Monitor	$189.00
C1660 Auto Modem	$59.99
DPS 1101 Daisy Printer	$339.00
Comrex 220 (C64 Interface)	$89.99

GRAPHICS

Polaroid

Palette	$1399.00
Power Processor	$229.00
Illuminated Slide Mounter	$39.99
Polacolor 2 Pack film	$18.99

ACCESSORIES

CURTIS POWER STRIPS

Diamond SP-1	$32.99
Emerald SP-2	$39.99
Sapphire SPF-1	$52.99
Ruby SPF-2	$59.99
Safe Strip SP-3	$19.99

KENSINOTON

MasterPiece	$99.99
MasterPiece Plus	$119.00

PORTABLE COMPUTERS

HEWLETT PACKARD

41CV	$139.00
41CX	$199.00
HP 11C	$49.99
HP 12C	$75.99
HP 15C	$75.99
HP 16C	$89.99
HPIL Module	$98.99
HPIL Cassette or Printer	$359.99
Card Reader	$143.99
Extended Function Module	$63.99
Time Module	$63.99

We etock the full line of NP celculetor producte

NEC

PC-8401 LS	CALL
PC 8201 Portable Computer	$319.00
PC-8231 Disk Drive	$599.00
PC-8221A Thermal Printers	$149.00
PC-8281A Data Recorder	$99.99
PC 8201-06 8K RAM	$79.99

SHARP

PC-1350	$149.00
PC-1261	$149.00
PC-1500A	$169.00
PC-1250A	$89.99
CE-125 Printer/Cassette	$129.00
CE-150 Color Printer Cassette	$149.00
CE-161 16K RAM	$129.00

SOFTWARE

ARTWORX

Bridger 4.0 (All)	$17.99
Strip Poker (All)	$19.99

ATARI

ROMS FOR ATARI ONLY

8036 Atari Writer	$24.99
Star Raiders	$4.99
Missile Command	$4.99
Defender	$4.99
Galaxian	$4.99
Asteroids	$4.99
Centipede	$4.99

BROOERBUND

Print Shop (All)	$29.99
Graphics Library I, II, III (All)	$17.99
Bank Street Writer (64 AT)	$34.99
Bank Street Writer (IIe/IIc)	$44.99

BATTERIES INCLUDED

Paperclip (AT, 64/128)	$37.99
Paperclip (IIe/IIc)	$44.99
Homepack (64/AT)	$29.99
D E G A S (All)	$27.99

CONTINENTAL SOFTWARE

Home Accountant (All)	$44.99
Tax Advantage (All)	$44.99
Home Acct /Tax Bundle (All)	$59.99

MICROPROSE

Kennedy Approach (All)	$24.99
Aerojet (All)	$24.99
Silent Service (All)	$24.99

PFS

File (64)	$39.99
File, Report, Graph (IIe/IIc)	$79.99
Write w/Proof (IIe/IIc)	$79.99

SPRINGBOARO

Newsroom (64)	$34.99
Clip Arts (64)	$22.99
Newsroom (Apple)	$39.99
Clip Arts (Apple)	$22.99

SUBLOGIC

Flight Simulator (All)	$37.99
Jet (64)	$37.99

DISKETTES

maxell

3½" SS/DD (10)	$24.99
3½" DS/DD (10)	$34.99
5¼" MD-1 w/Hardcases (10)	$12.99
5¼" MD-2 w/Hardcases (10)	$18.99
5¼" MD-2-HD for AT (10)	$39.99
3½" 5 pack SS/DD/Case	$13.99

Verbatim.

5¼" SS/DD	$12.99
5¼" DS/DD	$24.99
Disk Analyzer	$24.99

Dennison

Elephant 5¼" SS/SD	$13.99
Elephant 5¼" SS/DD	$14.99
Elephant 5¼" DS/DD	$16.99
Elephant Premium DS/DD(50)	$79.99
Elephant 3½" SS/DD	$29.99

IBM

5¼" DS/DD floppy disks (Box of 10)	$26.99

DISK HOLDERS

INNOVATIVE CONCEPTS

Flip'n File 10	$2.99
Flip'n File 50	$14.99
Flip'n File 50 w/lock	$19.99
Flip'n File 100	$19.99
Flip'n File Data Case	$9.99

AMARAY

50 Disk Tub 5¼"	$9.99
30 Disk Tub 3½"	$9.99

MODEMS

ANCHOR

Volksmodem	$59.99
Volksmodem 300/1200	$189.99
Signalman Express	$259.00
Lightning 2400 Baud	$399.00
Expressi	$199.00
6470 (64/128) 300/1200 Baud	$139.00

OIOITAL OEVICES

AT300 - 300 Baud (Atari)	$99.99

Hayes

Smartmodem 300	$139.00
Smartmodem 1200	$389.00
Smartmodem 1200B	$359.00
Smartmodem 2400	$599.00
Micromodem IIe	$149.00
Smart Com II	$89.99
Chronograph	$199.00
Transet 1000	$309.00

AST

Reach 1200 Baud Half Card	$399.00

SUPRA

MPP-1064 AD/AA (C-64)	$69.99

Novation

Smart Cat Plus	$299.00
J-Cat	$99.99
Novation 2400	$549.00
Apple Cat II	$229.00
212 Apple Cat II	$379.00
Apple Cat 212 Upgrade	$229.00
Macmodem	$279.00

QUADRAM

Ouadmodem II	
300/1200	$339.00
300/1200/2400	$499.00

EVEREX

1200 Baud Internal (IBM/PC)	$199.00

DRIVES

HARD

I-MEGA

A110H Bernoulli Single10	CALL
A210H Bernoulli Dual 10	CALL
A220H Bernoulli Dual 20	CALL
Save on Bernoulli Carts	CALL

TALLGRASS TECHNOLOGIES

25, 35, 50, 80 meg (PC)	from $1299.00

IRWIN

Tape Backup	CALL

EVEREX

60 Meg Internal Backup System	$799.00

CORE

AT20-AT72MB	CALL

PRIAM

40, 60 MB (PC, XT, AT)	CALL

KITS

10 Meg with controller	$379.00
20 Meg with controller	$519.00

FLOPPY

INDUS

Atari GT	$199.00
C-64 /128 GT	$199.00

MSD

SD1 C-64 Single	$219.00
SD2 C 64 Dual	$469.00

Tandon

320K 5¼" (PC)	$119.00

TEAC

320K 5¼"	$119.00

Dept. A204 CALL TOLL-FREE

MONITORS

AMDEK

Video 300 Green	$119.00
Video 300A Amber	$129.00
Video 310A Amber TTL	$159.00
Color 300 Composite	$169.00
Color 600 Hi-Res RGB	$399.00
Color 710 Ultra Hi-Res	$439.00
Color 722 Dual Mode	$529.00

MAONAVOX

8562 RGB/Composite	$279.00

NEC

JB1205A	$79.99
JB1270G/1275A	(ea.) $99.99
JB1280G TTL	$129.00
JB1285A TTL	$129.00
JC1460 RGB	$229.00
JC1225 Composite	$179.00
JC1401 Multi Sync RGB	$549.00

PRINCETON

MAX-12E Amber	$179.00
HX-9 9" RGB	$469.00
HX-9E Enhanced	$519.00
HX-12 12" RGB	$469.00
HX-12E Enhanced	$559.00
SR-12 Hi-Res	$599.00
SR-12P Professional	$699.00

TAXAN

115 12" Green	$119.00
116 12" Amber	$129.00
121 TTL Green	$139.00
122 TTL Amber	$149.00
610 510x200 RGB	$NEW
620 640x200 RGB	$NEW
630 640x200 RGB	$NEW
640 720x400 RGB	$NEW

QUADRAM

8400 Quadchrome I	$499.00
8410 Quadchrome II	$339.00
8420 Amberchrome	$179.00
8500 Quad Screen	$1449.00

ZENITH

ZVM 1220 Amber	$99.99
ZVM 1230 Green	$99.99
ZVM 1240 IBM Amber	$149.00
ZVM 135 RGB	$459.00
ZVM 1330 RGB	$459.00
ZVM 1360 RGB	CALL
ZVM 1380 E.G. Comp	CALL

INTERFACES

AST

Multi I/Q (Apple II)	$159.00

PRACTICAL PERIPHERALS

Graphcard	$79.99
Seriall Card	$99.99
Microbuffer II+	$169.00
Microbuffer 32K	$189.00

QUADRAM

Microfazer	from $139.00
Elezer (Epson)	from $79.99

Orange Micro

Grappler CD (C64)	$89.99
Grappler + (Apple)	$89.99
Grappler 16K + (Apple)	$159.00

DIGITAL DEVICES

Ape Face (Atari)	$49.99
U-Print A (Atari)	$54.99
U-A16/Buffer (Atari)	$74.99
U-Call Interface (Atari)	$39.99
U-Print C (C64)	$49.99
P-16 Print Buffer	$74.99
U-Print 16 apple IIc	$89.99

PRINTERS

Canon

A40,A50,A55	CALL
LBP-8A1 Laser	CALL

CITIZEN

MSP-10 (80 col.)	$279.00
MSP-15 (132 col.)	$389.00
MSP-20 (80 col.)	$349.00
MSP-25 (132 col.)	$509.00

C.ITOH

Prowriter 7500	$169.00
Prowriter 1550P	$349.00
Starwriter 10-30	$399.00
3500 Tri Printer	$1499.00

corona

Lazer LP-300	$2799.00

DIABLO

D25 Daisywheel	$549.00
635 Daisywheel	$899.00
D80IF Daisywheel	CALL

daisywriter

2000	$699.00

EPSON

Homewriter 10, LX-80	CALL
FX-85, FX-286, RX-100, JX-80	CALL
DX-10, DX-20, DX-35	CALL
SQ-2000, Hi-80, HS-80, AP-80	CALL
LQ-800, LQ-1000, LQ-1500	CALL

JUKI

6000 Letter Quality	CALL
6100 Letter Quality	CALL
6200 Letter Quality	CALL
5300 Letter Quality	CALL
5510 Dot Matrix	CALL

LEOENO

808 Dot Matrix 100 cps	$179.00
1000 Dot Matrix 100 cps	$259.00
1380 Dot Matrix 130 cps	$289.00
1385 Dot Matrix 165 cps	$339.00

NEC

3000 Series	$779.00
8000 Series	$1099.00
ELF 360	$399.00
Pinwriter 560	$999.00

OKIDATA

182, 183, 192, 193, 2410, 84	CALL
Okimate 10 (Specify C64/Atari)	$189.00
Okimate 20 (IBM)	CALL

Panasonic

KX1080	NEW
KX1091	$259.00
KX1092	$389.00
KX1592	$469.00
KX1595	$659.00

QUADRAM

Quadjet	$399.00
Quad Laser	CALL

SILVER-REED

500 Letter Quality	$279.00
550 Letter Quality	$419.00
800 Letter Quality	$699.00

star

SG-10A (Atari)	CALL
SG-10C (C64 Interface)	CALL
SB/SD/SG/SR Series	CALL
Powertype Letter Quality	CALL

Texas Instruments

TI850	$529.00
TI855	$639.00
TI865	$799.00

TOSHIBA

1340 (80 column)	$389.00
P341 (132 column)	$799.00
P351 (132 column)	$1049.00

SOFTWARE FOR IBM

ANEA EOFTWARE

Paradox	$549.00

ASNTON-TATE

Framework II	$389.00
dBase III Plus	$389.00

BORLANO

Lightening	$54.99
Sidekick (unprotected)	$49.99
Reflex	$54.99
Newspack	$54.99

CENTRAL POINT

Copy II PC-Backup	$29.99

DECISION REEOURCEE

Chartmaster	$229.00
Signmaster	$169.00
Diagram Master	$219.00

FIFTN OENERATION

Fast Back	99.99

FOX AND OELLER

Quickcode III	$169.00

FUNK EOFTWARE

Sideways	$44.99

NARVARD EOFTWARE INC.

Total Project Manager	$269.00

INFOCOM

Cornerstone	$279.00

LIFETREE

Volkswriter III	$159.00

LIVINO VIOEOTEXT

Think Tank	$109.00
Ready	$64.99

LOTUE

Symphony	CALL
1-2-3	CALL

MECA EOFTWARE

Managing Your Money 2.0	$99.99

MICROPRO

Easy	$94.99
WordStar 2000	$239.00
WordStar 2000 +	$289.00
WordStar Professional	$199.00

MICRORIM SOFTWARE

R Base 4000	$249.00
R Base 5000	$389.00
Clout 2.0	$129.00

MICROSOFT

Flight Simulator	$34.99
MultiPlan	$129.00
Word	$229.00
Mouse	$139.00

MICROSTUF

Crosstalk XVI	$89.99
Crosstalk Mark IV	$149.00
Remote	$89.99

MULTIMATE

Multi Mate Word Proc	$219.00
Advantage	$289.00
On File	$89.99
Just Write	$89.99

NOUNEMON

Intuit	$69.99

NORTON

Norton Utilities 3.1	$59.99

ONE ETEP

Golf's Best	$37.99

PFE: IBM

Proof	$59.99
File/Graph	(ea) $79.99
Report	$74.99
Write/Proof Combo	$79.99

PROFEEEIONAL SOFTWARE

Write-N-Spell	$89.99

TNE BOFTWARE OROUP

Enable	$329.00

EATELLITE EYETEME

Word Perfect 4.1	$219.00

EORCIM/IUE

Accounting AP/AR/GL/INV/OE	(ea.) $299.00
SuperCalc III	$199.00
EasyWriter II System	$199.00
Super Project	$199.00

EPI EOFTWARE

Open Access	$379.00

BUBLOGIC

Jet	$37.99

IBM

IBM PC SYSTEMS

Configured to your specifications.

Call for Best Price!

IBM-PC, IBM-XT, IBM-AT

ZENITH

PC-138 Series, PC-148 Series, PC-158 Series, PC-160 Series, PC-171 Series, AT 200 Series	CALL

SANYO

MBC 550-2, MBC 555-2, MBC 675 Portable, MBC775, MBC 880 Desktop	CALL

AT&T

Safari (7300)	CALL
6300	CALL

corona

PPC400 Dual Portable	$1289.00
PPCXT 10 meg Portable	$1989.00
PC40022 Dual Desktop	$1389.00
PC400-HD2 10 meg	$1989.00

ITT

ITT X TRA	
256K, 2 Drive System	CALL
256K 10 meg Hard Drive System	CALL
XP5, 20 meg	CALL

SPERRY

Sperry-AT	as low as $1749.00
Sperry-IT	as low as $2699.00

Call for Specific Configuration!

All Models	CALL

KAYPRO

KP 2000 Portable	CALL
Kaypro PC	CALL

MULTIFUNCTION CARDS

AST

Rampage	$379.00
Six Pack Plus	$229.00
I/Q Plus II	$139.00
Advantage-AT	$399.00
Graph Pak/64K	$599.00
MonoGraph Plus	$399.00
Preview Mono	$299.00
PC Net Cards	$379.00
5251/11 On-line	$669.00
5251/12 Remote	$579.00

dca

IRMA 3270	$879.00
IRMA Print	$999.00
IRMA Smart Alec	$779.00

EVEREX

Edge Card	$259.00
Graphics Edge	$239.00
Magic Card II	$169.00

HERCULES

Graphics	$299.00
Color	$159.00

IDEAssociates

IDEA 5251	$589.00

MYLEX

The Chairman	$439.00

PARADISE

Color/Mono Card	$149.00
Modular Graphics Card	$259.00
Multi Display Card	$219.00
Five Pack C, S	$129.00

PERSYST

Bob Board	$359.00

TECMAR

Captain - 64	$199.00
Graphics Master	$469.00

QUADRAM

Quadport-AT	$119.00
Liberty-AT (128K)	$349.00
The Gold Quadboard	$449.00
The Silver Quadboard	$239.00
Expended Quedboard	$199.00
Liberty	$309.00
QuadSprint	$499.00
QuadLink	$399.00
QuadColor	$199.00
Chronegraph	$79.99
Parellel Interface Board	$64.99

INTEL

PCNC8087 5MHz	CALL FOR YOUR PC
PCNC8087-2 8 MHz	
PCNC80287 6 MHz	
1010 PC-Above Board	
1110 PS-Above Board	
2010 AT-Above Board	

COMPUTER MAIL ORDER

1-800-233-8950 Dept. A204

Pressing CONTROL and HELP returns a value of 145. The statement POKE 732,0 clears location 732, so you can check for subsequent keypresses.

Apple IIe/IIc Compatibility

I'm interested in buying an Apple IIc computer. Can it use IIe hardware and software?

Carlos Aguayo

The Apple IIc computer is basically an Apple IIe that has been redesigned to take up as little space as possible. To keep the IIc small, Apple left out the IIe's expansion slots (where additional hardware can be attached), but added a built-in 5¼-inch disk drive. They also put the most common IIe expansion hardware (80-column video display, an extra 64K of memory, and two serial input/output ports) on the main board of the IIc. In addition, the IIc has some features that weren't available when the IIe appeared: an advanced 65C02 microprocessor and a character set called Mousetext *which contains extra characters especially for Macintosh-style icon- and menu-based programs. The newest version of the IIe (called Enhanced IIe) does have these extra features; dealers can upgrade an older IIe at a small cost.*

The IIc can run almost all IIe programs, as long as no special hardware is required. For instance, some music programs can communicate with instruments through a MIDI (Musical Instrument Digital Interface) adapter. This adapter must connect to an expansion slot, which is possible only on a IIe. Other programs sometimes expect a parallel I/O interface to attach a printer. Since the IIc has only serial *I/O, it can't run that type of modification. Although the IIc has no expansion slots, its peripherals (serial ports, disk drives, etc.) act like they are built into certain slots. Apple tried to select the most commonly used slot for each peripheral (printer in slot 1, disk drive in slot 6). However, not everyone puts everything in the same place, and some programs may demand an unconventional configuration. IIe owners can rearrange the cards in their slots to run such programs, but IIc owners don't have this option.*

The serial ports on the IIc generate standard RS-232 signals which can be used to communicate with most modems from any manufacturer. Many of the most popular printers are also available with RS-232 interfaces. But the IIc does not have standard connectors for these ports. To save space on the back panel of the computer, DIN-type connectors are used instead; as a result, you'll need special cables (available from Apple dealers) to attach serial peripherals.

When it comes to expandability, the IIe is much more flexible than the IIc. Almost any kind of peripheral can be attached through one of its slots, including parallel I/O ports, MIDI interfaces, hard disk drives, coprocessors, huge RAM expansion cards, and a host of other devices. However, some third-party companies have begun modifying the IIc to put in extras like additional memory and Z80 processors (to run the CP/M operating system, a popular IIe add-on). It's still more difficult than expanding a IIe, but it can be done.

IBM PUT And GET

I own a TI-99/4A and an IBM PCjr. Lately, I've been trying to convert some programs from TI to IBM. I have only one problem: the PUT and GET graphic statements in the IBM system. I really don't understand them. Could you show me a way to make an image and move it?

Billy Mobley

First, be aware that IBM BASIC has two types of GET and PUT statements: one for graphics and another for random files. The syntax for each type is different, so be sure you're using the graphics type. GET grabs the screen image within a specified rectangle and stores a copy of it in an array. PUT does just the opposite, putting the image from an array back onto the screen.

Several important rules apply to PUT and GET. Before using either command, you must be in a graphics mode (SCREEN 1, for example); neither PUT nor GET works on a text screen. The array that you GET a shape into must be a one-dimensional numeric array dimensioned to the proper size. Finally, you must GET before you can PUT.

The most difficult task is deciding what size to dimension the array. If the array is too small, it can't hold the graphics image, and the program won't work. The simplest solution is to try a large size like DIM A(500). It won't hurt to dimension it larger than necessary, but this method wastes memory. Here's a more efficient formula that tells you the minimum required size for the array:

INT((4+INT((x*res+7)/8)*y)/prec)

In this formula, the variable x *represents the width of the image in pixels;* y *is the height of the image;* res *is 1 for high resolution and 2 for medium resolution; and* prec *is the precision of the array (2 for integer, 4 for single precision, and 8 for double precision).*

GET must be followed by the screen coordinates of two opposite corners of the rectangular image, and the name of the array. For example, GET (0,0)–(19,29),A grabs a 20 × 30 pixel image at the top-left corner of the screen and stores it in array A. (Of course, you must first have an image on the screen. This can be done with DRAW.) With a high-resolution screen and a single-precision array, the formula above gives 23, so the dimension statement would be DIM A(23).

PUT is followed by the coordinates of the location on the screen where the top-left corner of the image is placed, then the name of the array, and an optional parameter for special effects. Five special effects are available: PSET, PRESET, AND, OR, and XOR. If no special effect is specified, XOR is assumed.

PSET displays the image exactly as it appeared when GET was used. PRESET displays a negative image. AND displays only those parts of the image that overlap an image already on the screen. OR superimposes the image onto an image already on the screen. XOR is a combination of AND and PRESET, reversing only those parts of the image that overlap an image already on the screen. The best way to understand exactly what these special effects do is to try them yourself. Using our example, PUT (200,100),A,PSET displays the image stored in the A array in the center of the screen.

The operation of XOR may seem strange, but it's handy for animation. When you PUT using XOR twice in the same position, the screen is restored unchanged. This allows you to move an image over a background image, giving a 3-D effect. Animation with XOR is a three-step process: PUT the image on the screen with XOR, calculate the new position, PUT the image in the old position a second time to erase it. By performing these steps repeatedly, the image seems to move. The following program moves a ball across the screen.

```
10 SCREEN 1
20 DIM A(113)
30 CIRCLE (20,20),20
40 PAINT (20,20)
50 GET (0,0)-(40,40),A
60 CLS
70 FOR C=1 TO 100
80 PUT (X1,Y1),A 'display ima
   ge
90 X2=X1+1:Y2=Y1+1 'calculate
    new position
100 PUT (X1,Y1),A 'erase imag
    e
110 X1=X2:Y1=Y2 'old=new
120 NEXT 'repeat
```

Simpler Absent Printer Test

I'm writing with regard to the "Readers' Feedback" item on absent Commodore printers, published in the December 1985 COMPUTE!. Another way to avoid a DEVICE NOT PRESENT error is to access the appropriate device (4) through the command channel (15) and check the value of the status variable ST. If ST does not equal 0, then the printer is not present. Here is a short routine to demonstrate:

```
10 OPEN 15,4,15:CLOSE 15
20 IF ST<>0 THEN 40
```

All the exciting, entertaining, and educational games, applications, and utilities from COMPUTE! magazine are now available on disk

for your Commodore, Atari, Apple, or IBM personal computer.

The COMPUTE! Disk

A new *COMPUTE! Disk* is published every month, rotating among the four major machines covered by *COMPUTE!*: Commodore 64 and 128; Atari 400/800, XL, and XE; Apple II-series; and IBM PC, PCjr, and compatibles.

Every three months you can receive a disk with all the quality programs from the previous three issues of *COMPUTE!* that will run on your brand of computer.

Like the popular *COMPUTE!'s Gazette Disk*, the *COMPUTE! Disk* is ready-to-load and error-free. It saves you valuable hours of typing time and eliminates typing errors.

With a subscription, you will receive one disk every three months for a total of four disks a year—for only $39.95. That saves you $20 a year off the single-issue cost.

Or you can order individual issues of the *Disk* for $12.95 a disk plus $2.00 shipping and handling.

Remember to specify your type of computer when ordering the *COMPUTE! Disk*. You'll find more information about this month's *COMPUTE! Disk* in this issue. (Note: You'll need the corresponding issues of *COMPUTE!* magazine to use the *Disk* since the disk will have no documentation.)

For fastest service when ordering a subscription to the *COMPUTE! Disk*, call toll free 1-800-247-5470 (in Iowa 1-800-532-1272).

For more details or to order individual issues of the *COMPUTE! Disk*, call our Customer Service Department toll free at 1-800-346-6767 (in New York 212-887-8525).

Please allow 4–6 weeks after placing an order for your first disk to arrive.

COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

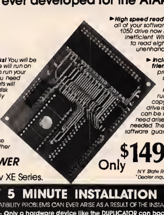
The 1050 DUPLICATOR IS HERE...

THE 1050 DUPLICATOR: The most powerful diskdrive copy system ever developed for the ATARI.

The Duplicator for The New "ST" is now available

The only Copy System You will ever need!

What will it do?



►***The main purpose of the Duplicator is to copy disks!*** *You will be able to copy just about any disk! The copies you make will run on any Atari drive. The Duplicator need not be present to run your backup copies. The Duplicator is fully automatic. You need only insert source and destination disks. Custom formats will be read and in turn reproduced on the backup copy disk. Our device will reproduce any custom format or heavily copy guarded scheme, bad sectors, double sectors, 19 through 24 sector format will present no problem to the Duplicator.*

►***You will still have single density, density and one half, and double density.*** *When you have a Duplicator installed in a 1050 drive that drive will be turned into true double density. You will have twice the disk storage. Your drive will be compatible with other double density drives as The Rana Indus Percom, etc.*

HARDWARE POWER

Fully Compatible with the XL & New XE Series.

►***High speed read & write.*** *Your disk drive will read and load all of your software, saving wear and tear on your drive. The 1050 drive now reads one sector at a time. This is slow and inefficient. With the duplicator installed you will be able to read eighteen sectors in the time it takes standard, unenhanced drives to read one.*

►***Included with every Duplicator will be user friendly disk software.*** *A simple, menu driven program will allow you to copy all of your software. A Duplicator enhanced drive will be a SMART drive. We plan to write many new and exciting programs that can only be run on an enhanced drive, eg. sending a copy-guarded disk over the phone. Since the drive is now fully programmable future upgrades can be made available to you on disks, should the need arise. No further hardware changes will ever be needed. The Duplicator comes with a full hardware and software guarantee.*

Only $149⁹⁵

Plus $2.50 for shipping handling
Add 7% outside U.S.A.
N.Y. State Residents add 7½% Sales Tax
**Dealer inquires are welcome. call for quantity price quote*

EASY 5 MINUTE INSTALLATION

NO HARM TO YOUR DRIVE OR INCOMPATIBILITY PROBLEMS CAN EVER ARISE AS A RESULT OF THE INSTALLATION OF OUR DUPLICATOR

IMPORTANT: **Only a hardware device like the DUPLICATOR can backup heavily copy-guarded disks. Don't be fooled by software programs that claim to do this.**

DUPLICATING TECHNOLOGIES inc.
Formerly Gardner Computing

99 Jericho Tpke., Suite 302A Jericho N.Y. 11753 Order Business Hrs **(516) 333-5805, 5807, 5808** Order Eve's and Weekends **(516) 333-5950**

TERMS We accept American Express, Visa, MasterCard and C.O.D. orders. Foreign orders must be in U.S. dollars. All personal checks allow 14 days to clear. Most items shipped within 24 hours.

```
30 OPEN 1,4:PRINT#1,"PRINTER I
      S ON":CLOSE 1:END
40 PRINT CHR$(147);"TURN ON PR
      INTER":GOTO 10
```

If you run this program with the printer off, it instructs you to turn the device on. Printing begins as soon as the printer is active.

Jim Plavecsky

Thanks for this compact, all-BASIC solution. In programs that open disk files or use an RS-232 device (usually a modem), you may want to perform this check at the very beginning, before you perform any other OPEN statements. The statement CLOSE 15 closes all other channels in addition to the command channel, terminating any RS-232 communications and disconnecting (but not really closing) any open disk files.

Emphasized TI Character Set

The custom character set given for the Commodore 64 on pages 108–109 of COMPUTE!'s January 1986 issue can be used on the TI-99/4A with only slight modifications. Since that character definition data is listed in hexadecimal format, it can be read as a *pattern-identifier* string and assigned with the CALL CHAR statement (see page II-76 in the *TI User's Reference Guide*). Each line in the Commodore character set listing contains data for a single character plus a checksum value at the end of the line. To convert the data in each line to a 16-character pattern-identifier string, type in the first eight two-digit hexadecimal numbers (spaces are left out, of course). In the first line, for instance, the resulting string could be used with a CALL CHAR statement to redefine the @ character. To create the new character set, first enter this program:

```
100 FOR L=1 TO 94
110 READ C$
120 CALL CHAR(L+32,C$)
130 PRINT CHR$(L+32);
140 NEXT L
150 GOTO 150
```

Next, you must enter a series of lines containing DATA statements. Each DATA statement represents the data for one character in the form of a 16-character pattern identifier string. For example, the first DATA line would look like this:

500 DATA 7CC6DEDEC0C07800

Here is how to enter all of the DATA lines.

lines	**500–800**	**Use data from line 7108–71F8 (defines ASCII characters 33–63)**
line	**810**	**Use data from line 7000 (ASCII 64)**
lines	**820–1070**	**Use data from lines 7208–72D0 (ASCII 65–90)**
line	**1080**	**Use data from line 70D8 (ASCII 91)**
line	**1090**	**Use 00C06030180C0600 as data (ASCII 92)**
lines	**1100–1110**	**Use data from lines 70E8–70F0 (ASCII 93–94)**
line	**1120**	**Use 00000000000000FF as data (ASCII 95)**
line	**1130**	**Use data from line 7200 (ASCII 96)**
lines	**1140–1390**	**Use data from lines 7008–70D0 (ASCII 97–122)**
lines	**1400–1420**	**Enter data from lines 72D8–72E8 (ASCII 123–125)**
line	**1430**	**Enter 000020745C080000 as data (ASCII 126)**

The result of your effort will be an emphasized font with true lowercase.

John Hedstrom

Thank you for your suggestion.

©

HOTWARE: Software Best Sellers

This Month	Last Month	Title	Publisher	Remarks	Apple	Atari	Commodore	IBM	Macintosh
Entertainment									
1.		*Jet*	SubLogic	Jet simulation			●		
2	4.	*Silent Service*	MicroProse	Submarine simulation	●	●	●	●	
3	3.	*Ultima IV*	Origin Systems, Inc.	Fantasy game	●	●	●		
4.	2.	*Karateka*	Brøderbund	Action karate game	●	●	●		
5.	1.	*F-15 Strike Eagle*	MicroProse	Air combat simulotion	●	●	●	●	
Education									
1.	1.	*Typing Tutor III*	Simon & Schuster	Typing instruction program	●		●	●	●
2	2	*Math Blaster!*	Davidson	Introductary math program, ages 6-12	●	●	●	●	
3.	3	*New Improved MasterType*	Scarborough	Typing instruction program	●	●	●	●	●
4.	4.	*Music Construction Set*	Electronic Arts	Music composition program	●	●	●		
5.	5.	*I Am The C-64*	Creative/ Activision	Introduction to the C-64			●		
Home Management									
1.	1.	*Print Shop*	Brøderbund	Do-it-yourself print shop	●	●	●		
2.	2.	*The Newsroom*	Springboard	Do-it-yourself newspaper	●		●	●	
3.		*Bank Street Writer*	Brøderbund	Word processor	●	●	●	●	
4.	3.	*Print Shop Graphics Library III*	Brøderbund	Upgraded graphics library	●	●	●		
5.	4.	*Print Shop Graphics Library*	Brøderbund	100 additional graphics	●	●	●		

Copyright 1986 by Billboard Publications, Inc. Compiled by the Billboard Research Department and reprinted by permission. Data as of 2/1/86 (entertainment) and 2/8/86 (education and home management).

IT ALSO RUNS ON 64K.

Serious runners know it takes more than great running shoes to improve performance. It takes knowledge. Now Puma gives you both. With the RS Computer Shoe. The first training shoe to combine advanced footwear technology with computer technology.

The RS Computer Shoe has a custom-designed gate array built into its heel. This computer chip records your run, then communicates the results to any Apple IIE, Commodore 64 or 128, or IBM PC computer.

A software program included with the shoe automatically calculates your time, distance and calories expended. Then graphically compares them to past performances and future goals.

The RS Computer Shoe from Puma. We're so out front in technology, we put computers in the backs of our shoes.

PUMA®
OUR WORD FOR QUALITY

Apple is a registered trademark of Apple Computer Inc., Commodore 64 and 128 are trademarks of Commodore Computer Systems. IBM and IBM PC are registered trademarks of IBM

NEW TECHNOLOGIES
The Converging Digital Universe

Selby Bateman, Features Editor

The digitization of America is well under way. Thanks to a wave of new consumer electronics products, this year more people than ever will see and hear how the convergence of digital audio, video, satellite, telephone, optical, laser, television, and computer technologies is transforming the world. Yet, the phenomenon is just beginning.

The winds of technological change have been blowing a gale for the past few years. And the forecast shows no indication of a letup. In fact, millions of consumers will begin to reap a resulting whirlwind of new high-tech products for the home, office, and classroom. Consider the following:

• A home stereo system answers your phone, takes messages, and alerts you to incoming calls.

• With the push of a button, your video film recorder captures a picture from your favorite TV show and instantly prints out a still photo for your wallet.

• Your 20-volume set of encyclopedias, contained and cross-indexed on a compact disc in a player connected to your computer, searches and prints out 37 reference sources on your selected topic in less than 30 seconds.

• The satellite dish in your backyard automatically tracks various communication satellites based on the pattern of TV programs you want to watch each night. At the same time, your computer is receiving and storing financial data that unobtrusively shares the same incoming satellite transmission to your TV.

• The digital TV in your living room displays two small windows on the screen while you watch a program uninterrupted; one window shows the changing stock quotations, while the second window displays a program from a different channel or previews a tape from your videocassette recorder.

• The computer image recorder connected to your personal computer makes a 35 mm slide, color print, or overhead transparency of the business chart or digital painting you've just created.

Does any of this sound far-fetched? You'll be able to buy products this year that do all of these things and more. If it seems difficult to keep up with the latest news about consumer electronics, it's not your fault. Never have so many dramatic technological changes produced so many new capabilities and products in so short a time. What has become strikingly clear is that all of these innovations share a common foundation—the digital, microprocessor-based world of computer electronics.

These changes have become so important to our lives and our pocketbooks that market researchers are now targeting a new group of consumers: Technologically Advanced Families (TAFs). Could "yuppies" eventually be surpassed in importance by "taffies," households that purchase and use the latest computers, VCRs, stereo TVs, 8 mm camcorders (camera recorders), compact disc players, satellite dishes, and dozens of other products? Consumer electronics manufacturers and retailers believe that these households are the important leading-edge market for their array of new products.

Among the catalysts sparking enthusiasm for the latest in high-tech gear, none is more important than the personal computer phenomenon of the past half-dozen years. Not only are computer owners the bedrock of the TAFs, but the new generation of 16/32-bit computers is powerful enough to work with just about any other consumer electronics product. Suddenly, devices like VCRs, compact disc players, electronic keyboards, and camcorders have become computer peripherals. As these products continue to become more sophisticated and flexible, their technologies converge and their capabilities expand. In the world of consumer electronics, the whole has indeed become more than the sum of its parts.

The development of the microcomputer has accelerated an already rapid evolution, says David Allen, president of Boston Media Consultants and a writer specializing in TV production, computers, videodiscs, and videotape. "They come along with greater speed. That's not a function of any interactivity, that's just a curve that the computer industry and microelectronics industry are on.

"Each development feeds the next development in a serendipitous way that makes succeeding developments faster to accomplish," says Allen. "You can really say that we're now to the point at which you could almost create any technological package you could conceive of, if you don't put a price restriction on it. Nothing is technologically impossible, in a broad sense. But it has to be accompanied by some kind of way to get return on investment. And that's what slows things down more than anything else right now. It's market-driven, not technologically driven."

During the past year, a parade of new technologies has entered the computer scene. The arrival of MIDI (Musical Instrument Digital Interface) has opened the doors to a new world of computer-based music composition and performance (see "Making Music with MIDI," COMPUTE!, January 1986). Laser-driven compact disc technology has branched out from stereo systems to computer data storage and retrieval. Smaller, less expensive video cameras and camcorders that connect with VCRs and computers are making inroads in consumer markets.

In addition, a new family of audio/video hardware and software products has been created to take advantage of the latest computers, particularly the Commodore Amiga, Atari ST, and Apple Macintosh.

It's appropriate that in this age of video one of the most promising fields of development is computer control of video images that originate from video cameras, VCRs, laser disc players, other computers, or TVs with video outputs—essentially any device that puts out a composite video signal. For instance, Commodore is releasing two fascinating video peripherals for the Amiga: the Genlock, which plugs into the back of the Amiga and mixes external video signals with the computer's own video output; and the Amiga LIVE digitizer (formerly known as the "frame grabber"), which captures and digitizes an external video image in the Amiga itself.

Commodore/Amiga's Genlock accessory tucks beneath the rear of the Amiga computer and permits sophisticated video image mixing.

"Genlock is external to the Amiga and externally mixes two video sources, one of them the Amiga's," explains Paul Higginbottom, an Amiga product manager at Commodore. "So you take the Amiga's video source and the external video source, and you combine them—and the audio as well. Nothing comes into the Amiga with Genlock. With Amiga LIVE, a digitized picture is brought into the Amiga. So one [Genlock] is doing superimposing, and the other [Amiga LIVE] is actually taking an image and bringing it in.

"They operate separately, but you could certainly use them together," says Higginbottom. "You may want to take a real image and put Amiga's graphics on it, and digitize those back into the Amiga again."

Immediate applications for the Genlock include on-screen titling for video presentations or home movies, "electronic chalkboard" effects similar to those used for TV sports analysis, and special video effects achieved by mixing Amiga graphics with other video images. At the Amiga's official unveiling in New York last summer, artist Andy Warhol used a video camera, Genlock, and Amiga LIVE to digitize a picture of rock singer Deborah Harry, then used a mouse-controlled graphics program to "paint" the video image with new colors. Amiga LIVE can be used not only for special video effects such as these, but also for video databases, says Higginbottom.

"We don't just mean pretty pictures. If you're a real estate agent or an architect, or you have a parts list you want to inventory, something like that—then you can have a video inventory," he explains. "And Amiga LIVE performs in realtime, not like most digitizers you see that usually take anywhere from 8 to 30 seconds to generate the picture on the screen. This is in realtime; if you have a movie camera, you'll see the image move as you move the camera."

Both the Genlock and Amiga LIVE are expected to be available in April or May, pending final FCC approval. Each accessory will cost about $249.95.

A different video digitizer is in the works for the Atari ST and should be available by the time you read this. Hippopotamus Software is introducing the Hippovision Video Digitizer this spring for the ST and plans to have a version available later for the Amiga. (No price announced yet.)

"Anything that produces video signals, you just plug into the [digitizer] box that's connected to the computer," says Clint Ballard, vice president of engineering for the Los Gatos, California firm. "You press a button when you get a picture you like, and there you have it. We'll also have image processing software with which you can change around the colors—do whatever you want with it. This really opens up the graphics world."

For the Macintosh, which has a two-year head start on the Amiga and ST, there are already several video digitizers and compatible graphics programs available. *MacVision* from Koala Technologies, *Micro-Imager* from Servidyne Systems, Inc., *Thunderscan* from Thunderware, Inc., and a few others

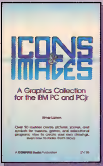
COMPUTE!'s IBM LIBRARY

COMPUTE! offers you special savings on this set of top-selling titles for your IBM PC and PCjr.

Each book contains valuable tutorials, programming guides, personal and business applications, and games. Together, the books provide all the up-to-date, ready-to-use information and programs you need to get the most from your IBM personal computer.

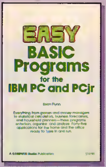
COMPUTE!'s First Book of IBM

Edited, 326 pages
Thirty of the best games, utilities, graphics and sound generators, and applications for the IBM PC and PCjr. A disk is also available which includes programs in the book, $12.95, 0106BDSK.
ISBN 0-87455-010-8
$14.95


ICONS IMAGES
A Graphics Collection for the IBM PC and PCjr
Elmer Larsen
A COMPUTE! Books Publication
$14.95

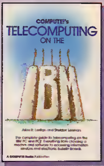
Icons and Images: A Graphics Collection for the IBM PC and PCjr

Elmer Larsen, 227 pages
Ninety-four short routines to instantly enhance business, educational, and entertainment programs on either the IBM PC or PCjr.
ISBN 0-942386-84-1
$14.95



Easy BASIC Programs for the IBM PC and PCjr

Brian Flynn, 359 pages
Everything from games to home and office applications programs is included for the IBM PC and PCjr.
ISBN 0-942386-58-2
$14.95


COMPUTE!'s
TELECOMPUTING
ON THE
IBM
Arlan R. Levitan and Sheldon Leemon
A COMPUTE! Books Publication

COMPUTE!'s Telecomputing on the IBM

Arlan R. Levitan and Sheldon Leemon, 274 pages
The ins and outs of telecomputing on the IBM PC or PCjr, from selecting a modem and evaluating terminal software to getting online with the major information services.
ISBN 0-942386-98-5
$14.95

ORDER ALL FOUR BOOKS FOR $49.95 AND SAVE *OVER 15%* OFF THE RETAIL PRICE!

Take advantage of the great price savings and exceptional value of these bestselling books from COMPUTE! and order the four-book set today.

To order, call our toll free customer service number, 1-800-346-6767 (in NY 212-887-8525) and ask for COMPUTE!'s IBM Library. Or, mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

You can also order individual copies of any of the above books at the stated retail price. Sorry, no substitutions. LIMITED TIME OFFER! You must order before May 20 to receive your 15-percent discount.

Please send me the books I have selected. My payment is enclosed.

_______ ***Sets of COMPUTE!'s IBM Library (four books per set) $49.95*** _______
_______ ***COMPUTE!'s First Book of IBM (010-6) $14.95*** _______
_______ ***COMPUTE!'s Telecomputing on the IBM (98-5) $14.95*** _______
_______ ***Easy BASIC Programs for the IBM PC and PCjr (58-2) $14.95*** _______
_______ ***Icons and Images: A Graphics Collection for the IBM PC and PCjr (84-1) $14.95*** _______

All orders must be prepaid
□ Payment enclosed (check or money order)
□ Charge □ MasterCard □ Visa □ American Express

Subtotal _______
NC residents add 4.5% sales tax _______
Shipping and handling
($2.00 per book, $5.00 per book airmail) _______
Total amount enclosed _______

Account No _______________ Exp Date _______________ (required)
Name _______________
Address _______________
City _______________ State _______________ Zip _______________
Please allow 4–6 weeks for delivery
36411911

COMPUTE!™ Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England and in Canada from Holt, Rinehart, & Winston, 55 Horner Avenue, Toronto, ON M8Z 4X6.

SC1224
ATARI

THE ST™ COMPUTER LINE FROM ATARI.®

IT'S LIKE GETTING THE POWER AND SPEED OF A FERRARI® FOR THE PRICE OF A FORD.®

When Atari introduced the 520ST™, we set the personal computer industry on its ear.

Nobody had ever produced a machine so powerful and technically advanced for such an incredibly low price. Nobody but Atari has done it yet.

The competition was stunned.

The critics wrote rave reviews.

And consumers were ecstatic.

We could have rested on our laurels, but we didn't.

Instead, Atari extended the ST concept to a new computer called the 1040ST™.

The amazing new 1040ST is even more powerful than the 520ST and years ahead of all the competition at almost any price. The only question in 1986 isn't which company to buy a computer from, but which computer to buy from Atari.

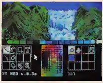
	ATARI® 1040ST*	COMMODORE® AMIGA™	IBM® PCAT™	APPLE® Macintosh™	APPLE IIc®
Price	$999	$1795	$4675	$1995	$1295
CPU Speed MHz	68000 8 0	68000 7 16	80286 6 0	68000 7 83	65002 1 0
Standard RAM	1 MB	256K	256K	512K	128K
Standard ROM	192K	192K	64K	64K	16K
Number of Keys	95	89	95	59	63
Mouse	Yes	Yes	No	Yes	Optional
Screen Resolution (Non-Interlaced Mode) Color Monochrome	 640x200 640x400	 640x200*** 640x200***	 640x200 720x350**	 None 512x342	 560x192 560x192
Color Output	Yes	Yes	Optional	None	Yes
Number of Colors	512	4096	16	None	16
Disk Drive	3 5"	3 5"	5 25"	3 5"	5 25"
Built-in Hard Disk (DMA) Port	Yes	No	Yes	No	No
Midi Interface	Yes	No	No	No	No
# of Sound Voices	3	4	1	4	1

Atari 520ST with 512K RAM, $799

*Connects to standard color TV For RGB color monitor add $200

**With optional monochrome board (non bit-mapped)

***Interlace Mode – 640x400



At $799, the 520ST gives you 512 Kbytes of RAM, a high-resolution monochrome monitor, 2-button mouse, and 3.5" disk drive.

At $999, the 1040ST gives you 1024 Kbytes of RAM, an ultra high-resolution monochrome monitor, 2-button mouse, and a built-in double-sided 3 5" disk drive, plus built-in power supply. Both the 520ST and the 1040ST can be connected directly to your own color T.V. Or you can add an Atari RGB color monitor to get the sharpest, most colorful images possible. Add $200 for color monitor.

It's simply a matter of choosing which model best fits your needs.

And whether you choose the 520ST or the 1040ST, you'll be getting the power and speed of a Ferrari for the price of a Ford.

In fact, you'll save hundreds and in some cases thousands of dollars over comparable computers. Which is why consumers are still ecstatic. Why the critics are still writing rave reviews. And why the competition is still stunned.

Ferrari is a registered trademark of Ferrari Italia SpA and Ferrari of America Inc Ford is a registered trademark of Ford Motor Company IBM and PCAT are registered trademarks of International Business Machines Corp Commodore and Amiga are trademarks of Commodore Electronics LTD Apple Apple IIc and Macintosh are trademarks of Apple Computer Inc Atari 520ST 1040ST and ST are trademarks of Atari Corp

ATARI®

POWER WITHOUT THE PRICE.™

make excellent use of the Mac's high-resolution monochrome graphics. Since the Amiga and the ST each boast superb color graphics as well as high-resolution modes surpassing the Mac's, video digitization hardware and graphics software are becoming even more flexible and powerful.

As computers grow more capable of handling video images, other manufacturers are gearing up to take advantage of new markets expected to develop. Toshiba and Polaroid have announced products which strengthen the connections among computers, photography, and video. The two companies are jointly introducing a new instant video film recorder that produces instant color prints or slides from a TV set or monitor and has optional RGB (red-green-blue) computer input. The recorder features digital freeze-field capture, color preview capability, and accepts standard NTSC (National Television Standards Committee) signals.

The recorder captures and digitizes any image from a TV screen, whether the signal originated from a broadcast station, VCR, video camera, or any other standard video device. When equipped with the appropriate camera, the result is an instant photo print or 35 mm slide. With the push of a button, you could freeze one frame of your home movies, your favorite rock video, or a TV show, and then instantly produce a color picture. The recorder is expected to be available by midyear.

Polaroid is also introducing this year an improved version of its Palette computer image recorder. The Palette provides presentation-quality photos from computer graphics generated by a wide variety of computers, such as the Apple II series and the IBM PC family. It's capable of handling image resolutions up to 920 × 700, depending on the combination of hardware and software. Almost all presentation-graphics and graphics-editing software is compatible with the under-$2,000 system.

Although few personal computer owners will spend several thousand dollars to buy such video systems for the home, the next few years will see dramatic price drops as technology improves and costs decline.

For example, Kodak's Consumer Electronics Division plans to introduce a still video system that allows you to select and record individual video images. The system's player/recorder captures images in realtime from any NTSC video signal and stores up to 50 images on a tiny floppy disk. An adjunct to this system is a film-to-disk transfer station that may be installed at film processors; you could have 35 mm color negatives transferred to the floppy disk, then view the pictures at home on your TV—ordering regular prints later, if you like.

Kodak had also planned to announce a new color video imager for producing instant prints of any video image. However, a recent decision by the U.S. Supreme Court on behalf of Polaroid has forced Kodak to withdraw from the instant photography business. Although Kodak had expected initial sales of the video imager to be in commercial and industrial applications, the long-range plan was to make the product part of home computer and video centers, according to Richard D. Lorbach, vice president of Kodak's consumer division.

"We anticipate that the color video imager eventually will be used as a home entertainment center component," said Lorbach before the court decision was handed down. "Our market research indicates that there is significant consumer interest in being able to make photographs of personal images displayed on TV screens."

This type of video system presents a wide range of possibilities. For example, by capturing images from your home videos, you could make a slide show of still shots or produce prints or slides for family albums. Computer artists could take their digital paintings or images captured from a video source and create their own sequenced video show. With the appropriate computer software, text could be overlayed on any of the images.

There are hundreds of business and industrial applications for this technology. Rather than spending thousands of dollars on outside production of sales and marketing presentations, almost any business would have access to high-quality video production. A real estate agency could take photos or videotapes of its properties, add textual information on prices and other details, and then show the resulting package to their customers. Any of the frames could be turned into glossy prints for the house-hunters to keep for reference. The ramifications are virtually limitless.

One of the most important developments in the marriage of computer and video technology is the introduction of digital TVs—TV sets that convert the incoming analog broadcast signal into digital form. Toshiba, Sony, and most of the other large consumer electronics companies have invested millions of dollars to develop digital TV. Exceptionally clear pictures are only one of the benefits of this research. Digital TVs also have what's called PIP (picture-in-picture) capability—they can partition the viewing screen by opening separate "windows" for simultaneously displaying other video signals.

An example is the 26-inch DT-2680A TV receiver/monitor from NEC Home Electronics. It can simultaneously display the picture from the station that's tuned in plus moving pictures from any of three auxiliary video inputs, or color computer graphics through the set's RGB input. You can watch two channels at once, or a channel and a videotape, or even work with your home computer while watching TV on the same screen.

The picture you'll be watching is much sharper, too. Today's conventional TVs offer approximately 250 lines of horizontal screen resolution, while the NEC digital TV is capable of resolving up to 500 lines. This is actually more resolution than is available from broadcast signals. Through special filtering, the digital TV displays a broadcast screen resolution of 336 lines—the best that's possible with today's broadcasts.

In addition, the NEC digital TV has enough microprocessor-based memory to store up to three different still video pictures at a time. By pressing a button on the remote control, you can capture any video image and display it as an 8½-inch

The Computer Book Club®

Recognized as the authoritative source for micro users!

Superior Performance, Practical Price!

More programs, projects, ways to use your micro for home, hobby, education, and business!

Select 5 Books for Only $2.95

Values to $123.75

1884
List $24.95

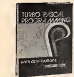
1000P
List $8.25 (paper)

1997
List $21.85

2648
List $22.95

1748P
List $10.85 (paper)

1522
List $25.85



2927
List $23.85

1873
List $21.95

1160
List $14.95

1960
List $19.95

1229P
List $8.25 (paper)

1907
List $21.85

1911
List $21.95

1750P
List $12.95 (paper)

1737
List $18.85

1979
List $25.85

1540
List $22.95

1718P
List $15.50 (peper)

1951
List $19.95

1831P
List $9.95 (paper)

1407
List $17.95

1978P
List $17.95 (peper)

1993
List $21.85

1845
List $18.85

1285P
List $10.25 (paper)

1992
List $18.85

1427P
List $12.95 (peper)

1275
List $15.85

Free guide to BASIC Statements & Commands

Plus FREE For Joining →

7 very good reasons to join The Computer Book Club®

- **Big Savings.** Save 20% to 75% on books sure to increase your computer know-how
- **No-Risk Guarantee.** All books returnable within 10 days without obligation
- **Club News Bulletins.** All about current selections—mains, alternates, extras—plus bonus offers. Comes 13 times a year with hundreds of up-to-the-minute titles to pick from
- **Automatic Order.** Do nothing, and the Main selection will be shipped automatically! But . . . if you want an Alternate selection—or no books at all—we'll follow the instructions you give on the reply form provided with every News Bulletin
- **Bonus Books.** Immediately get Dividend Certificates with every book purchased and qualify for big discounts of 60% to 80%
- **Extra Bonuses.** Take advantage of added-value promotions, plus special discounts
- **Exceptional Quality.** All books are first-rate publisher's editions selected by our Editorial Board and filled with useful up-to-the-minute information

The Computer Book Club®

P.O. Box 80, Blue Ridge Summit, PA 17214

Please accept my membership in Tha Computer Book Club® and send tha 5 volumes circled below, *plus* my FREE copy of *BASIC Statements, Commands and Functions* billing me only $2.95 plus shipping and handling charges. If not satisfied, i may raturn the books within tan deys without obligation end have my membership canceled. I agree to purchase 4 or more books at regular Club prices (plus shipping/handling) during tha next 12 months, and may resign any tima tharaaftar.

1000P 1160 1228P 1275 1295P 1407 1427P 1522 1540
1718P 1737 1748P 1750P 1811 1831P 1873 1878 1882 1884
1907 1945 1951 1960 1978P 1993 1997 2627 2648

Name________________________________

Address______________________________

City_________________________________

State__________ Zip__________ Phona__________

Valid for new members only. Foreign applicants will receive ordering instructions. Canada *must remit* in U.S. *currency*. This order subject to acceptance by The Computer Book Club®.

CMPT-486

(diagonal) window within the 26-inch screen. Meanwhile, the background video image is unaffected. You could freeze-frame a fullback plowing through the line while watching the play continue on the main screen.

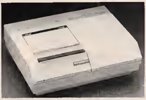
As might be expected, the connection capabilities and special features of such a TV set go far beyond the few video and audio plugs found on even the better current sets. The NEC digital TV contains a stereo amplifier and stereo speakers, three sets of line video inputs for VCRs, video disc players, color cameras, and home computers, and an eight-pin RGB input. Outputs include a monitor jack that carries whatever is on the screen, a TV output that carries whatever channel is tuned, external speaker outputs, fixed audio line outputs for recording, and variable audio line outputs for volume-controlled connections to an external sound system.

As NEC vice president Gerry Tangney says, this "is a taste of the future of home TV." The NEC digital set is expected to be introduced in May, with the price to be announced soon.

Another new technology already on the horizon is high-definition TV (HDTV), an enhanced broadcast signal that offers 1,125 scan lines of information instead of the 525 now used in conventional American TV broadcasting. This would require broadcasters to upgrade their equipment, however, and efforts to adopt an HDTV standard have reportedly been mired in international and corporate disagreements over how to bring about this doubling of screen clarity.

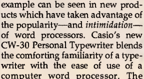
The growing popularity of compact disc (CD) audio players has given new impetus to the development and widespread consumer distribution of their digital data cousins, called CD-ROMs (Compact Disc-Read Only Memories). Although these laser discs are only 4.72 inches in diameter, they are capable of storing 600 megabytes of information on a single side, with an access time of seconds.

The first company out the door with CD-ROM players in the retail market is the Subsystems and Peripherals Division of North American Philips Corporation. Its CM 100 disc player and CM 155 controller card works with the IBM PC-compatible computers (other interfaces will be announced this year). Available with the Philips CD-ROM player is Grolier's *The Electronic Encyclopedia*, the equivalent of a 20-volume reference collection on just about a quarter of one side of a CD-ROM disc. Although the initial purchase price of $1,495 may keep initial sales out of the home market in volume, the price for CD-ROM technology is expected to drop quickly over the next couple of years.



Philips has introduced its CD-ROM drive which comes with Grolier's Electronic Encyclopedia *on a compact disc. The entire package sells for $1,495.*

Technology occasionally moves in mysterious ways, and an example can be seen in new products which have taken advantage of the popularity—and *intimidation*—of word processors. Casio's new CW-30 Personal Typewriter blends the comforting familiarity of a typewriter with the ease of use of a computer word processor. The $399.95 hybrid machine looks very much like a standard electric typewriter. But a quick look at the keyboard also shows a set of cursor and special function keys, plus a 15-character liquid-crystal display window for editing.



This Casio computer-compatible electronic typewriter is a hybrid—part typewriter and part word processor—that can connect to a computer to serve as a printer.

One of the most interesting features of the Casio typewriter is that it's computer-compatible. It contains both a Centronics-standard parallel interface and an RS-232 serial interface that lets the typewriter become a computer printer (plain or thermal paper). It can be hooked up to a 300 baud modem for uploading and downloading text with a computer. It has built-in pica and elite pitches, right justification, and multiple type fonts: boldface, underlining, double-wide characters, special symbols, and foreign alphabet characters. It has enough memory to store two pages of text, and with an optional memory expander, up to ten pages of text. Small removable memory cards let you save and store text. Casio obviously hopes to capture the best of both worlds, typewriters and word processors, at the same time it is attracting those who don't want to give up typewriters, but are fearful they're being left behind by word processors.

The Magnavox VideoWriter is an $800 dedicated word processor aimed at the home market.

Magnavox has taken a different approach with its new Videowriter, a dedicated home word processor that contains its own software, printer, spelling checker, and 18-line monitor (smaller than a regular computer screen, but larger than most portable computers). The $800 Videowriter has a memory capacity of approximately 70 pages of text, automatically stored on standard 3½-inch disks. While dedicated word processors have been used in offices for years, it's unusual to see such a product for the home market, especially considering the number of people who buy multipurpose computers primarily for word processing.

Computers are converging with yet another technology, too—telephones. For example, Commodore is planning to introduce its new 1100 AnswerMate, a programmable computer-controlled telephone answering machine for the Amiga. The AnswerMate connects to the Amiga's RS-232 port and to a telephone. Not only does it play back your taped greetings and record messages, but it also can respond with messages generated by the Amiga's built-in synthesized voice. And multitasking software included with the AnswerMate lets it answer phone calls while you're busy using the computer for other things. (Price to be announced.)

Commodore's AnswerMate connects to the Amiga computer to serve as a telephone answering machine that can make use of the Amiga's multiprocessing and synthesized speech capability.

There is scarcely an area of consumer electronics which is not moving either directly or indirectly toward the personal computer, either as a peripheral or as a microprocessor-based stand-alone device. Even the ways in which computer users receive their software may be undergoing change in the future.

For example, Cauzin Systems, with backing from Kodak, has developed the Softstrip system of information storage. Data is encoded on a strip of paper in a format similar to—but more compact than—the familiar bar codes found on consumer products. One strip, which typically measures 9½ by ⅝ inches, can store up to 5,500 characters (about three typewritten pages). The strips can be printed on ordinary paper and are read by an electro-optical scanner. Connected to a computer, the scanner reads the coded strips and transfers the data into memory for later storage on disk.

Further examples of converging electronics technologies abound in virtually every field. The emergence of stereo TVs and VCRs, coupled with a stereo-capable computer such as the Amiga, obviously opens new possibilities for audiophiles. Interactive video, spurred by improvements in laser discs, is another rapidly evolving technology with a connection to personal computing. Radio signals relayed by satellites can carry data accessible by computer users. Use of electronic mail systems is expected to jump from less than a billion messages a year today to more than 20 billion by the end of the decade, ultimately becoming a major service as common as the telephone and the U.S. mails.

As media consultant David Allen noted earlier, technology is capable of virtually anything today; but the successful marketing of an idea is the key to its success. In the forseeable future, neither technology nor the marketplace shows any signs of slowing down. ©

Attention Programmers

COMPUTE! magazine is currently looking for quality articles on Commodore, Atari, Apple, and IBM computers (including the Commodore Amiga and Atari ST). If you have an interesting home application, educational program, programming utility, or game, submit it to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Or write for a copy of our "Writer's Guidelines."

J&R MUSIC WORLD

23 PARK ROW, NEW YORK, N.Y. 10038

ORDER TOLL FREE 800·221·8180 IN NEW YORK, ALASKA & CANADA CALL: (212) 693 0396

Shop at home and save 10% to 50% and more ... on quality merchandise —fully guaranteed

SONY
SONY HIGH PERFORMANCE 5¼" DOUBLE DENSITY DISKETTES

Single Sided 10-pack	$16.95
Double Sided 10-pack	$19.95
3.5" Single Sided 10-pack	$24.95

ATARI
ATARI 520ST COLOR COMPUTER SYSTEM
•512K RAM •3.5" Micro Floppy Drive •12" Color Monitor •GEM operating system •FREE Software
$749.95

"TOP TEN" COMPUTER HARDWARE

ATARI 520ST MONOCHROME SYSTEM 512K drive monitor mouse	$649.95
COMMODORE 128/1571 PACKAGE 128K computer & new 1571 disk drive	$499.95
ATARI 130XE WORD PROCESSOR 128K 1050 drive 1027 printer software	$379.95
IBM XT COMPATIBLE COMPUTER 256K 2-360K drives monitor	$1299.95
HAYES SM-300 UNIVERSAL MODEM External RS 232 software	$139.95
HAYES SM1200B IBM MODEM Internal 1200 baud Auto answer dial	$319.95
ANCHOR 6470 C64/128 MODEM 300/1200 Baud Auto answer dial	$139.95
COMMODORE C-1902 13" MONITOR Color RGB composite hi-res	$319.95
XTRON IA-12X 12" MONITOR IBM compatible Amber with stand	$99.95
TAXAN 220 14" MONITOR Color/Green screen composite	$179.95

"TOP TEN" COMPUTER PRINTERS

STAR MICRONICS SG-10C DOT MATRIX Built-in C64 I/O interface/120cps	$239.95
JUKI 6100 LETTER QUALITY Triple pitch daisy wheel	$379.95
EPSON JX-80 FOUR COLOR PRINTER Colorful graphs tables charts	$369.95
BROTHER M1509 DOT MATRIX/NLQ 180 cps friction/tractor	$359.95
EPSON LB-1000 FAST DOT MATRIX 136 column [illegible]	$749.95
ATARI 1027 LETTER QUALITY Bi-directional friction feed	$129.95
EPSON LX-80 DOT MATRIX 100 cps NLQ mode 1K buffer	$219.95
LEGEND 808 FRICTION/TRACTOR [illegible] 100 cps [illegible]	$149.95
CANON A-50 DOT MATRIX 160 cps 2K buffer NLQ mode	$339.95
OKIDATA 192P MICROLINE 150 cps bit image graphics	$349.95

"TOP TEN" IBM COMPATIBLE SOFTWARE

dBASE III PLUS Ashton-Tate Networking relational DBM (ASH DBASE3+)	$369.95
REFLEX Borland Data base merge w/1-2-3 store (BOR REFLEX)	$59.95
TURBO LIGHTNING Borland On-line spelling verifier (BOR LIGHTNING)	$59.95
Q & A (512k Req) Symmantic Data base word proc. intelligent (SYM Q&A)	$239.95
LOTUS 1-2-3 VER 2.0 Lotus Devel. New improved version (LOT 123)	$339.95
JET Sub Logic New flight simulator (SUL JET)	$34.95
SCENERY DISKS 1-6 Sub Logic 6 new scenes Flight Sim. (specify #) Each	$14.95
APBA BASEBALL Random House Statistical recreation (RH BASEBALL)	$59.95
MULTIMATE 3.3 Multimate Intl. Full featured word processor (MTI MULTIMATE)	$259.95
PFS SERIES Software Publishing Choose write file or report (SPI specify) Each	$79.95

TOP TEN COMPUTER ACCESSORIES

MIDLAND ROSS 6100 SUPRESSOR 6 outlets circuit breaker 5' cord	$19.95
WICO COMMAND CONTROL JOYSTICK 2 fire buttons steel shaft	$24.95
RING KING RK1200 PRINTER STAND Fully adjustable all printers	$15.95
DISCWASHER SPIKEMASTER PROTECTOR 4 outlet 5 stage filtering	$54.95
CARDCO G-WIZ C64 PRINT/INTERFACE Interface with graphics support	$49.95
RING KING C50 5 1/4" DISK HOLDER Stores up to 50 disks high impact plastic	$10.95
RECOTON C2004 DISK DRIVE CLEANER 5¼" drive cleaner fluid reusable	$13.95
BUSH CT-120 SMALL COMPUTER DESK 28" H x 26" W x 18" D	$59.95
NASHUA 5 ¼" FLOPPY DISKETTES Single sided/double density 10-pack	$10.95
COMPUSERVE STARTER KIT 5 hours of access time any computer	$24.95

"TOP TEN" ATARI SOFTWARE

KARATEKA Broderbund Martial arts graphic adventure (BRO KARATEKA)	$24.95
TAX ADVANTAGE Continental 1985 version (CNE TAXADVANTAGE)	$44.95
PAPERCLIP Batteries Included Full featured word processor (BIA PAPERCLIP)	$34.95
SYNFILE PLUS Synapse Advanced data base (SYN SYNFILE)	$34.95
FLIGHT SIMULATOR II Sub Logic Various reports graphics (SLA FLIGHT)	$34.95
BASEBALL Micro League Real stat graphics 20 teams (MLA BASEBALL)	$24.95
MOVIE MAKER Electronic Arts Animated graphics (EAA MOVIEMAKER)	$24.95
ATARIWRITER PLUS Atari New version spell check (ATA AX2034)	$29.95
SARGON III CHESS Hayden Multi-level program (HAS SARGON3)	$34.95
F-15 STRIKE EAGLE Micro Prose Combat flying simulator (MPA F15)	$22.95

"TOP TEN" COMMODORE 64 SOFTWARE

HARDBALL Accolade Advanced graphic simulation (ACI HARDBALL)	$24.95
THE NEWSROOM Springboard Graphics with processing (SSN NEWSROOM)	$34.95
SWIFT TAX Timeworks 1986 edition avail through 4/85 (TIM SWIFTAX)	$34.95
PAPERBACK WRITER Digital Solutions Word processor 80 column (SRI PAPERBACK)	$29.95
ULTIMA IV Origin Software Ultima series advanced (TAO ULTIMAIV)	$39.95
ANIMATION STATION Suncom Graphics tablet disk (SUN ANIMATION)	$59.95
SILENT SERVICE Micro Prose Submarine simulation (MPO SILENT)	$24.95
PRINTSHOP Broderbund Create cards banners etc (BRJ PRINTSHOP)	$27.95
COMPUTER AMBUSH SSI Advanced war simulation (SMS AMBUSH)	$34.95
TYPING TUTOR III Simon & Schuster Monitor your progress (SIS TYPING3)	$29.95

"TOP TEN" ATARI 520ST AND COMMODORE 128 SOFTWARE

VIP PROFESSIONAL VIP Tech Spreadsheet DBM 520ST (VIB VIPPROFESS)	$139.95
KINGS QUEST II Sierra Animated adventure 520ST (ONS KQ2)	$34.95
HABAWRITER ST Haba Systems Word processor 520ST (HAS HABAWRITER)	$49.95
HIPPO-C Haba Systems C language compiler (HBS HIPPOC)	$49.95
MIND FOREVER VOYAGING Infocom Advanced text adventure 520ST (INS MIND)	$29.95
SUPERBASE 128 Precision Relational data base 128 (PSR SUPERBASE)	$59.95
WORD WRITER 128 Timeworks 80 column only windows (TMB WORDWRITER)	$44.95
PAPERCLIP Batteries Included New all 128 facts speller (BIC PAPERCLIP)	$59.95
NEVADA COBOL Commodore CP/M mode compiler 128 (CDR NCOBOL)	$49.95
PAPERBACK PLANNER 128 Digital Sol. High level spreadsheet 80 col (DRE PLANNER)	$39.95

"TOP TEN" APPLE SOFTWARE

PINPOINT Pinpoint Software On-line desktop manager (PP PINPOINT)	$39.95
CATALYST VER 3.0 Quark Disk program loader (QUF CATALYST3)	$59.95
MOEBIUS Origin Software Martial art graphic adventure (EAP MOEBIUS)	$37.95
PRINTSHOP COMPANION Broderbund Font editor printshop (BRP PRINTCOMPAN)	$24.95
HALLEY PROJECT Mindscape Solar system simulation (MOP HALLEY)	$29.95
SPELLBREAKER Infocom Expert level text adventure (INP SPELLBREAK)	$29.95
MANAGING YOUR MONEY Meca Tobias' finances mgmt (MVP MANAGEMONEY)	$119.95
GRAPHICS EXPANDER Springboard 300 graphics Printshop (SBP EXPANDER)	$24.95
BARDS TALE Electronic Arts Role playing adventure (EAP BARDS)	$27.95
BETTER WORD PROCESSOR Springbrd Better working series (SPP BETTERWORD)	$34.95

ULTIMA IV
Quest of the Avatar
•the long awaited fantasy role-playing milestone. Thats 16 times bigger than Ultima III •Available for Commodore 64 Atari and Apple II family
$39.95

TIMEWORKS
SWIFTCALC 128
•Powerful, easy to use, to understand •Designed for home or small business use •Prints your spreadsheet sideways •For Commodore 128
$49.95

Remember this is only a sampling of our stock of computers, software, peripherals and accessories. If what you need is not listed please call or write for prices and availability

ALL SOFTWARE DISK-BASED UNLESS OTHERWISE NOTED

NOTE: WE CARRY A FULL LINE OF AUDIO, VIDEO, AND COMPUTERS IN EVERY MAJOR BRAND. THIS LISTING IS JUST A SMALL SAMPLE IN OUR $10,000,000 INVENTORY. PLEASE CALL US TOLL FREE OR WRITE TO US FOR PRICES ON ITEMS NOT LISTED IN THIS AD

ALL PRODUCTS CARRY U.S. WARRANTIES
ALL DEFECTIVE PRODUCTS EXCHANGED WITHIN 30 DAYS

HOW TO ORDER BY MAIL: FOR PROMPT AND COURTEOUS SHIPMENT, SEND MONEY ORDER, CERTIFIED CHECK, CASHIER'S CHECK, MASTERCARD, VISA (include card number, interbank No., expiration date and signature). DO NOT SEND CASH. PERSONAL AND BUSINESS CHECKS MUST CLEAR OUR BANK BEFORE PROCESSING. $25 MINIMUM ORDER. Shipping, Handling & Insurance Charge is 5% of Total Order with a $3.95 minimum. [illegible] CONTINENTAL U.S., ALASKA, HAWAII, PUERTO RICO, AND CANADA ONLY. [illegible] SORRY, NO C.O.D.'S. NEW YORK RESIDENTS please add sales tax. ALL MERCHANDISE SHIPPED BRAND NEW, FACTORY FRESH, AND 100% GUARANTEED. WE ARE NOT RESPONSIBLE FOR ANY TYPOGRAPHICAL ERRORS

CUT OUT AND MAIL

ALL NEW FOR 1986

J&R MUSIC WORLD

3 FREE CATALOGS

•AUDIO/VIDEO/COMPUTER ELECTRONICS CATALOG
•RECORD, CASSETTE AND CD CATALOG
•VIDEO MOVIE CATALOG

IF YOU HAVE NOT ALREADY RECEIVED OUR LATEST CATALOGS, PLEASE FILL OUT COUPON AND MAIL OR CALL TOLL FREE 1-800-426-6027 AND WE'LL SEND THEM OUT IMMEDIATELY

PLEASE CHECK WHICH CATALOGS YOU WANT

□ AUDIO VIDEO COMPUTER □ RECORD CASSETTE □ VIDEO MOVIE

23 PARK ROW, DEPT. C4, NYC, NY 10038

NAME

ADDRESS

CITY STATE ZIP

PLEASE SEND $1.00 FOR SHIPPING AND HANDLING

CUT OUT AND MAIL

Report From:

The Winter Consumer Electronics Show

A Turning Point For Atari?

Tom R. Halfhill, Editor

Following up its strong showing at the Comdex computer show in November, Atari introduced a more powerful version of its ST at the Winter Consumer Electronics Show in January. Thanks to increasing sales, growing software support, widening distribution, and hints of new enhancements to come, industry watchers are suddenly taking more notice of Atari's bid for a comeback. Meanwhile, Commodore also entered 1986 with encouraging sales and Apple is responding with an improved Macintosh and lower prices.

A year ago it seemed impossible. Commodore founder Jack Tramiel had split with his successful computer company after a management dispute, bought the debt-ridden Atari that he had nearly destroyed in price wars, installed his sons in key positions, laid off most of the work force, rushed the design of a powerful 16/32-bit machine in only six months, introduced it at an unheard-of low price, and announced he was going to resurrect Atari as a major contender in the personal computer marketplace.

Atari still isn't home free. But the house that Jack built suddenly seems a lot more solid.

Strengthened by encouraging sales of the 520ST—according to estimates, at least 100,000 units worldwide through Christmas—Atari is now attracting more attention within the industry. "Frankly, a lot of people didn't think Jack would make it this far," says one observer. "Now they're taking him a lot more seriously."

The new Atari 1040ST is the first one-megabyte computer for under $1,000. It has 1,024K of RAM and a built-in, double-sided disk drive.

If Atari's comeback ultimately succeeds, the six-week period between late November 1985 and early January 1986 may well become recognized as the turning point. During that period, Atari piled up sizeable holiday sales and made impressive appearances at two crucial industry trade shows: Fall Comdex and the Winter Consumer Electronics Show. Both are held annually in Las Vegas and are among the largest trade shows in the U.S., with upwards of 100,000 people at each event.

At Comdex, which is oriented toward business computing, Atari demonstrated to skeptics that its 520ST was a real machine with dozens of software packages. At CES, a show that encompasses every consumer electronics product imaginable, Atari was the only major computer manufacturer in attendance and made three important announcements: the new 1040ST, a more powerful version of the 520ST with one megabyte of memory and a built-in disk drive; price reductions of $100 for the monochrome and color 520ST systems; and a shift to mass-market outlets such as department stores for the 520ST.

Atari's appearances at Comdex and CES seemed all the more impressive due to the conspicuous absence of its closest competitor, Commodore. People were surprised when Commodore missed Comdex because the company has been trying to position the Amiga as a business computer and Comdex was the ideal place. But there was shock when Commodore bowed out of CES because Commodore has never missed a CES since the days when wristwatches and calculators were its stock in trade.

Commodore didn't have much to say about missing the shows. However, one Commodore executive admitted he was "uneasy" about the reaction at CES—ironically, the rumors of imminent financial catastrophe that once followed Atari were now being whispered about Commodore. The rumors proved untrue, however, and Commodore says it definitely plans to attend the Summer CES in Chicago this June.

Actually, Commodore finished 1985 with heavy sales of its own. According to reliable estimates, Commodore sold about one million 64s, as many as 500,000 Commodore 128s, and at least 20,000 Amigas. Even Commodore was caught off guard by the 64 and 128 sales. In fact, insiders say Commodore tried twice during the fall to discontinue the 64, but had to restart production both times to meet sudden demand. As an indication that Amiga sales are healthy, the leading independent software supplier for the computer—Electronic Arts—says it recovered all of its 1985 Amiga development costs within two weeks after releasing its first Amiga products.

Although Atari and Commodore are still struggling financially, both have survived a rough market in 1985 and appear to be in better shape for 1986.

Since Atari was the only major computer company exhibiting at CES (IBM and Apple routinely avoid this show), most of the computer news was Atari-related. The main event was the introduction of the 1040ST, the first one-megabyte computer selling for under $2,500. In fact, it's the first one-megabyte computer selling for under $1,000.

The 1040ST is basically an enhanced 520ST and is fully compatible with existing ST software and hardware. The keyboard and all interfaces are identical: RS-232 serial and Centronics-standard parallel ports; in/out MIDI (Musical Instrument Digital Interface); floppy and hard disk interfaces; plus monochrome and analog RGB monitor outputs. The graphics-oriented user interface, GEM (Graphics Environment Manager), is the same.

New features on the 1040ST include one megabyte (1,024K) of Random Access Memory (RAM), twice the amount that comes with the 520ST; a built-in, double-sided 3½-inch disk drive with a capacity of 770K per disk, twice the capacity of the drive sold with the 520ST; a TV output jack; and an internal power supply for both the computer and drive, reducing the familiar clutter of external cables.

Like recent-model 520STs, the 1040ST comes with its Tramiel Operating System (TOS) in Read Only Memory (ROM) chips, freeing up more than 200K RAM that used to be required when loading TOS from disk. Also like the 520ST, the 1040ST comes with ST BASIC, the *NEOchrome* graphics-drawing program, and a word processor, *1st Word* (the 520ST comes with *ST Writer*). Atari says *1st Word* has GEM features such as drop-down menus and on-screen type fonts. (*ST Writer*, by contrast, is a direct translation from the *AtariWriter* word processor for eight-bit Ataris.)

There are two different 1040ST packages. With a high-resolution monochrome monitor, the suggested retail price is $999.95. With an analog RGB color monitor, the price is $1,199.95. Atari says the 1040ST will be sold only through computer dealers and should be available immediately.

The 520ST also underwent some minor changes. The latest models will be shipped with TOS in ROM and a TV output jack. To widen distribution, the 520ST will be sold through mass-market outlets in three different ways. A system that includes the computer, a single-sided 3½-inch disk drive (380K capacity), and hi-res monochrome monitor will now be priced at $699 suggested retail. The same system with an analog RGB color monitor instead of the monochrome screen will be priced at $899. Both prices are $100 lower than before. The 520ST components are also available separately: $399 for the computer, $199 for a single-sided drive, $299 for a double-sided drive, $199 for the monochrome monitor, and $299 for the RGB monitor.

Atari's 520ST has been improved with a ROM-based operating system, a TV output jack, and a $100 lower price. Also, for the first time, the computer and other components will be available separately in mass-market outlets.

Rumors abounded at CES about new developments for the ST line, including a better graphics chip, a bit-block transfer chip similar to the one in the Amiga, a 5¼-inch disk drive adapter for use with an IBM PC emulator, and more. Officially, Atari won't confirm or deny if it's preparing to introduce any of these products in the near future.

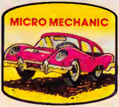
However, Atari is expected to announce at least one enhancement at an upcoming computer show in West Germany (about half of all ST sales are in Europe). The most likely possibility is the bit-block chip, which allows faster screen graphics. Also, it is now known that three companies outside Atari are working on PC emulators for the ST, each taking a slightly different approach. At this writing (mid-January), none of them is expected to be ready for several months.

Atari's CES exhibit poked fun at the Amiga and Apple Macintosh by lining up a series of computers running the now-famous Amiga bouncing ball demo. (A screen photo of this demo appears in COMPUTE!'s cover story on the Amiga in September 1985; it shows a red-and-white checkered globe spinning and bouncing around the screen, casting a transparent shadow on the background.) An Amiga, 520ST, Macintosh, and eight-bit Atari 130XE spent four straight days dribbling checkered balls at the Atari booth. Oddly enough, the 130XE version was perhaps the most impressive of all. Not only was the 130XE bouncing a checkered ball, but also a 3-D image of the Atari logo decorated with 128 rippling colors.

Apple hasn't been oblivious to the competition, and a week after CES announced an improved version of the Macintosh and lower prices for the 512K Mac. The new Macintosh Plus has one megabyte of RAM, a double-sided disk drive that stores 800K (twice the capacity of existing Macintosh drives), a redesigned keyboard with numeric keypad and cursor keys, a faster operating system, and an extra peripheral port called the Small Computer System Interface. The suggested retail price is $2,599. The 512K Mac was reduced from $2,499 to $1,999, and kits are available so owners of 128K and 512K Macs can upgrade to the Mac Plus.

FREE SOFTWARE

Now Get Up To 200 FREE Programs* When You Subscribe to *COMPUTE!* Today.



Subscribe to ***COMPUTE!*** today and you'll be getting a lot more than just another computer magazine. That's because ***COMPUTE!*** comes complete with up to 20 FREE programs in each big issue.

Subscribe now and you can depend on a steady supply of high quality, fun-filled programs like Cash Flow Manager, Speed Ski, Turtle Pilot, Boggler, Text Plot, Retirement Planner, and hundreds of other educational, home finance, and game programs the entire family can use all year long.

The free programs alone are worth much more than the low subscription price. But there's more to ***COMPUTE!*** than just free programs.

COMPUTE!'s superb articles deliver the latest inside word on everything from languages to interfaces...programming to disk drives. And our up-to-the-minute software reviews are must reading for any home user.

Whether you're a novice or an experienced user, ***COMPUTE!*** is perfect for you. So subscribe today. Return the enclosed card or call 1-800-247-5470 (in Iowa 1-800-532-1272).

COMPUTE! Publications,Inc. abc
Part of ABC Consumer Magazines Inc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books, and COMPUTE!'s Apple Applications

*Ranging from short techniques and programming tips to complete, powerful applications programs.

Atari didn't ignore its older products at CES. Among other things, two new eight-bit computer packages were unveiled.

The 130XE, a 400/800/XL-compatible computer with 128K RAM that was introduced last year, will now be available in a $399 package that includes a 1050 disk drive, 1027 printer, and five pieces of software: *AtariWriter*, *Music Painter*, *Paint*, *Star Raiders*, and *Silent Butler*. The 130XE is still available separately for $149.

Relics from another age? Nope. Videogame machines are still selling so well that Atari has redesigned the nine-year-old 2600 and introduced a new model, the 7800.

Atari's new lowest-end computer is the 65XE, a 400/800/XL-compatible machine with 64K RAM. It was actually announced at last year's Winter CES, but was withheld from the market until existing inventories of 800s and 800XLs were sold out. The 65XE will be sold separately for under $100 or in a package similar to the 130XE's for $300–$350.

Much to everyone's surprise, Atari also came to CES with a redesigned 2600 videogame machine (formerly known as the VCS) and the new 7800 game machine. What's that, you thought videogame machines were dead? Guess again. Atari says it sold over a million 2600s in 1985. Apparently many of them are going to new markets overseas—including, we hear, the People's Republic of China.

The revamped 2600 is smaller, lighter, sleeker, has a carrying handle, and costs less than $50. The 7800 machine was one of the last projects of the old Atari before Tramiel took over, and is now being unpacked from mothballs. Thanks to a chip named MARIA, the 7800's graphics are superior even to those of the eight-bit computers, and the machine accepts all 2600 cartridges without an adapter. It sells for less than $80 and comes with a new version of *Pole Position*, the hit car racing game.

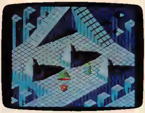
Computers aren't much good without software, and Winter CES demonstrated that a lot of companies are bringing out new programs for the ST and Amiga as well as popular eight-bit machines. Unfortunately, there were a lot fewer software companies at this CES than last year's. The West Hall, a large building which supplements the main Convention Center, once was filled with software publishers. This year, only a handful of them shared space with companies selling satellite dishes, videotapes, cable TV accessories, and other peripheralia. The last two years have been tough, and many software developers either can't afford to exhibit at CES anymore or are out of business altogether.

Still, some fascinating products are on the way. Starting with ST and Amiga software, here's what's new:

Electronic Arts (San Mateo, California) announced several new programs for the Amiga and its first products for the Atari ST. New Amiga software slated for release this year includes *Deluxe Music*, a note-oriented composition program; *Instant Music*, a composition program for nonmusicians; *Arcticfox*, a strategic combat tank simulation; *Adventure Construction Set*, for do-it-yourself adventure games; and *Deluxe Printing Construction Set*.

A previously announced Electronic Arts program, *Deluxe Video Construction Set*, is due for release soon. It lets you create animated sequences that can be integrated with screens created in *DeluxePaint*, a drawing program released in December. (Electronic Arts says it

Arcticfox *is a new strategic tank game for the Amiga from Electronic Arts.*



This is the Amiga version of Marble Madness, *but Electronic Arts is also bringing out an Atari ST version of the game.*

Deluxe Music *is a note-oriented composition program designed for the Amiga by Electronic Arts.*

Even if you don't know much about music, you can play songs on an Amiga with Electronic Arts' Instant Music *program. It has numerous built-in instrument sounds and doesn't rely on standard musical notation.*

NRI Trains You At Home—As You Build Your Own IBM-Compatible Computer

GET THE KNOW-HOW TO REPAIR EVERY COMPUTER ON THIS PAGE.

Learn the Basics the NRI Way—and Earn Good Money Troubleshooting Any Brand of Computer

The biggest growth in jobs between now and 1995, according to Department of Labor estimates, will occur in the computer service and repair business, where demand for trained technicians will actually *double*.

You can cash in on this opportunity—either as a fulltime corporate technician or an independent service-person—once you've learned all the basics of computers the NRI way. NRI's practical combination of "reason-why" theory and "hands-on" building skills starts you with the fundamentals of electronics, then guides you through advanced electronic circuitry and on into computer electronics. You also learn to program in BASIC and machine language, the essential languages for troubleshooting and repair.

You Build—and Keep—a 16-bit Sanyo personal computer

The vital core of your training is the step-by-step building of the 16-bit Sanyo MBC-550 series computer. Once you've mastered the details of this state-of-the-art machine, you'll be qualified to service and repair virtually every major brand of computer, plus many popular peripheral and accessory devices.

With NRI training, you learn at your own convenience, in your own home. You set the pace—without classroom pressures, rigid night-school schedules, or wasted time. You build the Sanyo IBM compatible computer from the keyboard up, with your own personal NRI instructor and the complete NRI technical staff ready to answer your questions or give you guidance and special help whenever you need it.

Praised by critics as the "most intriguing" of all the IBM-PC compatible computers, the new Sanyo uses the same 8088 microprocessor as the IBM-PC and features the MS/DOS operating system. As a result, you'll have a choice of thousands of off-the-shelf software programs to run on your completed Sanyo.

Your NRI course includes installation and troubleshooting of the "intelligent" keyboard, power supply, and disk drive, plus you'll check out the 8088 microprocessor functions, using machine language. You'll also prepare the interfaces for future peripherals such as printers and joysticks.

IBM is e Registered Trademerk of International Business Mechine Corporation.

Epson is e Registered Trademark of Epson America, Inc.

Apple end the Apple logo ere Registered Trademerks of Apple Computer, Inc

Compeq is e Registered Trademerk of COMPAQ Computer Corporation.

©1984 AT&T Technologies, Inc.

AND MORE.

100-Page Free Catalog Tells More

Send the coupon today for NRI's big 100-page color catalog on NRI's electronics training, which gives you all the facts about NRI courses in Microcomputers, Robotics, Data Communications, TV/Video/Audio Servicing, and other growing high-tech career fields. If the coupon is missing, write to NRI Schools, 3939 Wisconsin Avenue, NW, Washington, DC 20016.

Your NRI course includes the Sanyo 16-bit iBM compatible computer with 128K RAM, monitor, doubie density/doubie sided disk drive, end "intelligent" keyboerd; the NRI Discovery Leb®, teeching circuit design end operations; e Digitei Multimeter; Bundled Spreed Sheet end Word Processing Softwere worth over $1000 et reteil—end more.

NRI SCHOOLS
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue, Washington, DC 20016
We'll give you tomorrow.

For Career courses approved under GI bill,
□ check for details

☑ **CHECK ONE FREE CATALOG ONLY**

- □ **Computer Electronics with Microcomputers**
- □ Data Communications
- □ Robotics & Industrial Controls
- □ Video Electronics Servicing
- □ Electronic Design Technology
- □ Digital Electronics
- □ Satellite Communications
- □ Communications Electronics
- □ Industrial Electronics
- □ Basic Electronics
- □ Telephone Servicing
- □ Small Engine Servicing
- □ Appliance Servicing
- □ Automotive Servicing
- □ Air Conditioning, Heating, Refrigeration, & Solar Technology
- □ Building Construction
- □ Locksmithing & Electronic Security

Name (Please Print) ____________________ Age ______

Street ____________________

City/State/Zip ____________________

Accredited by the National Home Study Council

198-046

shipped 15,000 copies of *DeluxePaint* the first two weeks.) Scores composed with *Deluxe Music* can also be integrated with *Deluxe Video Construction Set.*

ST software from Electronic Arts includes two programs already released for the Amiga: *Financial Cookbook*, a home financial planner, and *Marble Madness*, an arcade game. Prices for Electronic Arts' Amiga and ST software range from $39.95 to $79.95.

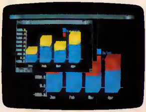
Hippopotamus Software (Los Gatos, California) is bringing out the Hippo ST Sound Digitizer and Hippovision Video Digitizer for the Atari ST. The sound digitizer plugs into the printer port and lets you sample, modify, and play back any type of sound. It includes special-effects software and a microphone for $139.95. The video digitizer lets you capture images in 256 × 256-pixel resolution from any composite video source, such as a video camera, videocassette recorder, videodisc player, or TV tuner. Picture files are compatible with Atari's *NEOchrome* drawing program, can be printed on color printers, and can be transmitted via modem. The price was not announced.

Hippopotamus has 13 other ST programs scheduled for release soon, including *HippoWord*, an $89.95 word processor; *HippoConcept*, an idea processor, $89.95; *HippoSimple*, a database manager, $49.95; *Hippo Disk Utilities*, $49.95; *HippoBackgammon*, $39.95; *HippoSpell*, a spelling checker with 30,000-word dictionary and user-definable terms, $39.95; *Hippo RAMdisk*, $34.95; *Hippo Computer Almanac*, which contains over 35,000 facts on everything from area codes to sports trivia, $34.95; *Hippo Jokes & Quotes*, with selectable PG, R, or X ratings, $34.95; *HippoArt I*, a collection of 30 picture files compatible with *NEOchrome*, $39.95; *Hippo EPROM Burner*, for programming your own chips, $139.95; *HippoClean*, a disk drive cleaning kit, $29.95; and *HippoPixel*, a utility for creating your own sprites and fonts, $39.95.

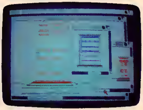
This strikingly beautiful picture was created on the Amiga's 320 × 200-pixel screen in 32 colors with Aegis Images, *a graphics-art program demonstrated at CES.*

Aegis Development (Santa Monica, California) is bringing out four graphics products for the Amiga: *Aegis Images*, a drawing program; *Aegis Draw*, a Computer-Aided Design (CAD) program; *Aegis Animator*, for creating animated sequences; and *Impact*, an executive graphics package. *Images, Animator*, and *Impact* were actually developed by Island Graphics of Sausalito, California. Under an agreement with Commodore, they were supposed to be released under the Amiga brand name. For instance, *Images* was originally known as *Graphicraft* or *ProPaint* (several screens created with this program appear on the Amiga's packaging and in the September 1985 issue of COMPUTE!). But Island Graphics and Commodore had a falling out, and the Amiga *Graphicraft* currently being sold is not the software developed by Island Graphics. Instead, Aegis acquired the marketing rights to the Island Graphics programs, enhanced them, and renamed them *Aegis Images, Animator*, and *Impact*.

Images is available for $79.95 separately, or for $139.95 in a package with *Aegis Animator*. Using *Images* screens as a backdrop, *Animator* allows 3-D animation and rotation, metamorphic shape manipulation, storyboarding of up to nine separate sequences, and ghost-line animation. *Impact* ($199.95) is for business presentation graphics and includes a slide show feature—charts, graphs, and pictures can be flipped in a predetermined sequence and transformed.

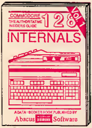


Impact, *another product from Aegis Development, makes it possible to produce three-dimensional business graphics using the Amiga's 4,096 colors.*



Aegis Draw *is a Computer-Aided Design (CAD) program for the Amiga which is aimed at professional users.*

C-128™ and C-64™ LIBRARY


COMMODORE 128 Vol.1
THE AUTHORITATIVE INSIDERS GUIDE
INTERNALS
A DATA BECKER BOOK PUBLISHED BY
Abacus Software

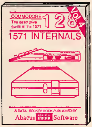
Detailed guide presents the 128's operating system, explains graphic chips, Memory Management Unit, 80 column graphice and commented ROM listings **500pp** $19.95

COMMODORE 128 Vol.2
Complete BASIC 7.0 ROM listings
BASIC 7.0 INTERNALS
A DATA BECKER BOOK PUBLISHED BY
Abacus Software

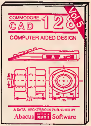
Get all the inside information on BASIC 7.0 This exhaustive handbook is complete with commented BASIC 7.0 **ROM** listings. Coming Summer '86 $19.95

COMMODORE 128 Vol.3
Practical & easy-to-use techniques for C-128
TRICKS & TIPS
BEST SELLER
A DATA BECKER BOOK PUBLISHED BY
Abacus Software

Filled with info for everyone. Covers 80 column hi-res graphics, windowing, memory layout, Kernal routines, sprites, software protection, autostarting **300pp** $19.95


COMMODORE 128 Vol.4
The descr ptive guide of the 1571
1571 INTERNALS
A DATA BECKER BOOK PUBLISHED BY
Abacus Software

Insiders' guide for novice & advanced users Covers sequentiel & relative files, & direct access commands. Describes DOS routines. Commented listings **500pp** $19.95



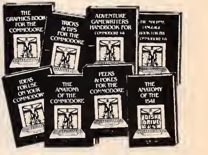
Learn fundamentals of CAD while developing your own system Design objects on your screen to dump to a printer Includes listings for '64 with Simon's Basic. **300pp** $19.95

Introduction to programing, problem analysis, thorough description of all BASIC commands with hundreds of examples, monitor commands, utilities, much more **400pp** $19.95

COMMODORE 128 Vol.7
Useful programming quick hitters
PEEKS & POKES
A DATA BECKER BOOK PUBLISHED BY
Abacus Software

Presents dozens of programming quick-hitters. Easy and useful techniques on the operating system, stacks, zero-page, pointers, the BASIC interpreter and more. $19.95

Essential guide for everyone interested in CP/M on the 128. Simple explanation of the operating system, memory usage, CP/M utility programs, submit files & more $19.95


THE GRAPHICS BOOK FOR THE COMMODORE
TRICKS & TIPS FOR THE COMMODORE
ADVENTURE GAMEWRITERS HANDBOOK FOR COMMODORE 64
THE MACHINE LANGUAGE BOOK FOR THE COMMODORE 64
IDEAS FOR USE ON YOUR COMMODORE
THE ANATOMY OF THE COMMODORE
PEEKS & POKES FOR THE COMMODORE
THE ANATOMY OF THE 1541
DISK DRIVE

ANATOMY OF C-64 Insider's guide to tha '64 internals. Graphics, sound, I/O, kernal, memory maps, more. Complete commented ROM listings. **300pp** **$19.95**

ANATOMY OF 1541 DRIVE Best handbook on floppy explains all. Many examples and utilities. Fully commanted 1541 ROM listings **500pp** **$19.95**

REVISED & EXPANDED

MACHINE LANGUAGE C-64 Learn 6510 code write fast programs. Many samples and listings for complete assembler, monitor, & simulator. **200pp** **$14.95**

GRAPHICS BOOK C-64 - best reference covere basic and advanced graphics. Sprites, animation, Hires, Multicolor, lightpen, 3D-graphics, IRO, CAD, projections, curves, more. **350pp** **$19.95**

TRICKS & TIPS FOR C-64 Collection of easy-to-use techniques: advanced graphics, improved data input, enhanced BASIC, CP/M, more. **275pp** **$19.95**

1541 REPAIR & MAINTENANCE Handbook describes the disk drive hardware. Includes schematice and techniques to keep 1541 running. **200pp** **$19.95**

ADVANCED MACHINE LANOUAGE Not covered elsewhere: - video controller, interrupts, timere, clocks, I/O, real time, extended BASIC, more. **210pp** **$14.95**

PRINTER BOOK C-64/VIC-20 Understand Commodore, Epson-compatible printers and 1520 plotter. Packed: utilitles; graphics dump; 3D-plot; commented MPS801 ROM listings, more. **330pp** **$19.95**

SCIENCE/ENGINEERINO ON C-64 In depth intro to computers in science. Topics: chemistry, physics, biology, astronomy, electronics, others. **350pp** **$19.95**

CASSETTE BOOK C-64/VIC-20 Comprehensive guide; many eample programs. High speed operating system fast file loading and saving. **225pp** **$14.95**

IDEAS FOR USE ON C-64 Themes: auto expenses, calculator, recipe file, stock lists, diet planner, window advertising, others. Includes listings. **200pp** **$12.95**

COMPILER BOOK C-64/C-126 All you need to know about compilers: how they work; designing and writing your own; generating machine code. With working example compiler. **300pp** **$19.95**

Adventure Oamewriter's Handbook Step-by-step guide to designing and writing your own adventure games. With automated adventure game generator. **200pp** **$14.95**

PEEKS & POKES FOR THE C-64 Includes in-depth explanations of PEEK, POKE, USR, and other BASIC commande. Learn the "inside" tricks to get the most out of your '64. **200pp** **$14.95**

Optional Diekettee for booka For your convenience, the programe contained in each of our books are available on diskette to save you time entering them from your keyboard. Specify name of book when ordenng. **$14.95 each**

C-128 and C-64 are trademarks of Commodore Business Machines.

Abacus Software

P.O. Box 7211 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $10.00 per book. Dealer inquires welcome–1200+ nationwide.

Aegis Draw, for hobby and professional CAD work, actually consists of two programs: *Aegis Draw* and *Aegis Draw Professional*. The first version retails for $199.95 and can be upgraded to the *Professional* version at extra cost. *Aegis Draw* has zooming, automatic scaling, selectable grids, layering, and multiple windows so you can work on several drawings simultaneously, or on different parts of the same drawing. It supports Kurta and Summagraphics digitizers, and plotters by Roland, Hewlett Packard, Houston Instruments, Epson, and Comrex. All four Aegis programs for the Amiga should be available immediately.

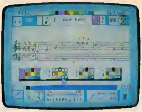
Supra Corporation (Albany, Oregon) announced a series of hard disk drives for the Atari ST with capacities of 10, 20, 40, and 80 megabytes. Prices start at $799 for 10 megabytes. They should be available this spring.

Optimized Systems Software (San Jose, California) was demonstrating *Personal Pascal* for the ST at the Atari booth and has already started shipping. This language supports all of the ST's special features, compiles and links most programs in about a minute (even faster with a hard disk or RAM disk), and sells for $74.95. OSS is also readying a version of the Prolog language and a set of disk utilities for the ST.

Cardco (Wichita, Kansas), known in the past for its VIC-20 and Commodore 64 add-ons, is preparing a one-megabyte memory expansion board for the Amiga. The board plugs into the expansion bus, and Cardco says it will be available this spring for about $400.

Unison World (Berkeley, California) is converting *PrintMaster*—a printer utility similar to *The Print Shop*—to the Atari ST. It's already available for the Commodore 64, IBM, and CP/M computers.

Activision (Mountain View, California) said sales of its initial software for the ST and Amiga have been "quite strong" and that additional titles will be released this year. These include *Garry Kitchen's GameMaker: The Computer Game Design Kit* and *The Music Studio* for the Amiga and ST; and *The Activision Little Computer People Discovery Kit* for the Amiga. *GameMaker* lets you write videogames without learning a programming language. Finished games can be saved on a blank disk and run independently of the master program, so you can distribute copies to friends. Activision is sponsoring a *GameMaker* contest—first prize is a trip to Activision plus $5,000.

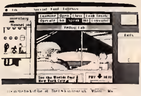


Another new music-composition program is The Music Studio *from Activision. There are versions for the Amiga, Atari ST, and IBM PCjr/Tandy 1000 computers.*

The Music Studio is a composition tool designed by Audio Light for both amateur and professional musicians. With it, you can design instruments and create new sound effects. The Amiga version plays up to 16 simultaneous sound channels in stereo.

Accolade/FTL Games (Cupertino, California) has released *Sundog: Frozen Legacy* for the ST. Originally written for the Apple II series, *Sundog* is a graphics strategy game that has been completely redesigned to take advantage of the ST's advanced graphics. Thanks to a proprietary data-compression scheme, hundreds of different full-color screens are stored on the program disk.

Mindscape (Northbrook, Illinois) is introducing three programs for the Amiga and one for the ST. *Brataccas* ($49.95) is a graphics adventure game for both computers that was developed by Psygnosis Limited of England. It was written specifically to take advantage of the 68000 chip inside the Amiga and ST. In *Brataccas*, you're a scientist who has invented a genetic process for creating a superbeing. With an evil government and the underworld in pursuit, you flee to a colonized asteroid, Brataccas. The object is to expose the government's corruption and clear your name. *Brataccus* is populated with nearly 60 different characters.



Mindscape's Deja Vu: A Nightmare Comes True *is a 1940s-style mystery game for the Amiga and Macintosh.*

For the Amiga only, Mindscape is releasing *The Halley Project: A Mission in Our Solar System* ($49.95), a realtime simulation of the solar system developed by Tom Snyder Productions with help from the Massachusetts Institute of Technology; *Deja Vu: A Nightmare Come True* ($54.95), a 1940s-style mystery game; and *Keyboard Cadet* ($39.95), a typing tutor. (Incidentally, Mindscape is the company which wrote the *Amiga Tutor* supplied with the Amiga.)

Abacus Software (Grand Rapids, Michigan) is importing a professional-quality program called *PC Board Design* for the ST. When Abacus finishes translating this circuit-designing utility from German, it will sell for $395.

Batteries Included (Richmond Hill, Ontario) was demonstrating its *D.E.G.A.S.* drawing program for the ST with a slideshow of screens called up in rapid sequence from a hard disk. *D.E.G.A.S.* started shipping just before CES, and it's already a hit—Batteries Included says that sales figures for the first two weeks were greater than for any other program in its history. ST and Amiga versions of the *Isgur Portfolio System*, a stock-management program, are scheduled for release later this year at $249.95.

Q-R-S (Buffalo, New York), a company that started back in 1900 by making music rolls for player pianos, is releasing its digital music library for the Amiga and ST. This consists of a number of disks containing piano music by Joplin, Gershwin, Liberace, and other artists and composers. Each disk contains six songs and sells for $19.95.

Sierra On-Line (Coarsegold,

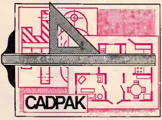
'128™ and C-64™ SOFTWARE

Our BASIC Compilers are the *complete* compiler and development packages. Speed up your programs from 5x to 35x.

Our BASIC Compilers give you many options: flexible memory management; choice of compiling to machine code, compact p-code or a mixture of both. Also on the '128, 40 or 80 column monitor output and FAST-mode operation.

The '128 Compiler's extensive 80-page programmer's guide covers compiler directives and options, two levels of optimization, memory usage, input/output handling, 80 column hi-res graphics, faster, higher precision math functions, speed and space saving tips, more. A great package that no software library should be without.

BASIC 128 Compiler $59.95
BASIC 64 Compiler $39.95



CADPAK is a remarkably easy to use drawing package for accurate graphic designs.

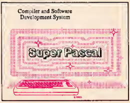
Using CADPAKs new dimensioning features you can create exact scaled output to all major dot-matrix printers.

This enhanced version of CADPAK allows you to input via the keyboard or a high quality lightpen. Two graphic screens permit you to COPY from one screen to another. DRAW, LINE, BOX, CIRCLE, ARC, ELLIPSE are but a few of the many selections to choose from. FILL objects with preselected PATTERNS; add TEXT; SAVE and RECALL designs to/from disk. You can define your own library of intricate symbols/objects with the easy-to-use OBJECT MANAGEMENT SYSTEM–it will store up to 104 separate objects.

C-128 $59.95
C-64 $39.95

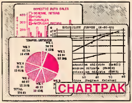
For schoof or software development, select **SUPER C**. Learn to use one of today's most popular languages on your Commodore 128 Powerful screen editor, compiler, linker–link up to seven modules, I/O library–includes printf and fprintf. Compile your C programs into *fast* machine language.

C-128 $79.95
C-64 $79.95



Not just a compiler, but a complete system for developing applications in Pascal. Extensive editor with search, replace, auto, renumber, etc. Standard J & W compiler that generates fast machine code. If you want to leam Pescal or to develop software using the best tools available–**SUPER Pescel** is your first choice.

C-128 $59.95
C-64 $59.95



For the professional who wants to easily create high quality charts and graphs without programming. You can immediately change the scaling, labeling, axis, bar-filling, etc. to suit your needs. Accepts data from CalcResult and MultiPlan. C-128 version has 3X the resolution of the '64 version. Outputs to most printers.

C-128 $39.95
C-64 $39.95

OTHER TITLES AVAILABLE:

Technical Analysis System

A sophisticated charting and technical analysis system for serious investors. By charting and analyzing the past history of a stock, TAS can help pinpoint trends & patterns and predict a stock's future. Enter data from the keyboard or from online financial services. **C-64 $59.95**

Personal Portfolio Manager

Complete portfolio management system for the individual or professional investor. Easily manage your portfolios, obtain up-to-the minute quotes and news, and perform selected analysis. Enter quotes manually or automatically through Warner Computer Systems. **C-64 $39.95**

Xper

XPER is the first "*expert system*" for the C-128 and C-64. While ordinary data base systems are good for reproducing facts, XPER can derive knowledge from a mountain of facts and help you make expert decisions. Large capacity. Complete with editing and reporting. **C-64 $59.95**

PowerPlan

One of the most powerful spreadsheets with integrated graphics for your Commodore computer. Includes menu or keyword selections, online help screens, field protection, windowing, trig functions and more. Power-Graph, the graphics package, is included to create integrated graphs & charts. **C-64 $39.95**

C-128 and C-64 are trademarks of Commodore Business Machines

Abacus Software

P.O. Box 7219 Grand Rapids, MI 49510 - Telex 709-101 - Phone (616) 241-5510

Call **now** for the name of your nearest dealer. Or to order directly by credit card, MC, AMEX of VISA call **(616) 241-5510**. Other software and books are available–Call and ask for your **free** catalog. Add $4.00 for shipping per order. Foreign orders add $12.00 per item. Dealer inquires welcome–1200+ nationwide.

California) has converted its series of 3-D graphics adventure games for the ST: *King's Quest I, King's Quest II,* and *Walt Disney's The Black Cauldron.* The *King's Quest* games have been particularly popular on IBM computers.

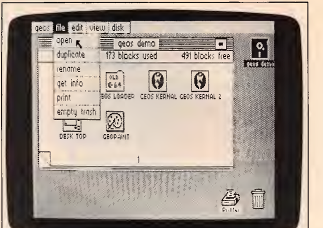
Spinnaker Software (Boston, Massachusetts) has converted *Fahrenheit 451, Treasure Island, Perry Mason, Nine Princes of Amber, Amazon,* and *Homework Helper Math* for the ST.

The popular eight-bit computers weren't ignored by software publishers, either, and some significant new programs were announced for the Commodore 64, 128, Apple, IBM PC/PCjr, and Atari. There isn't room here to list them all, but additional information can be found in this month's "News & Products" section.

One of the biggest hits at the show was a graphics-oriented operating system in final stages of development for the Commodore 64. Called *GEOS* (Graphic Environment Operating System), it's modeled after the user interface popularized by the Macintosh and adapted by the Atari ST and Amiga. *GEOS* loads from disk and replaces the 64's normal operating system. It speeds up disk access by a factor of five to seven times and displays a desktop screen with pull-down menus, icons, and windows. You manipulate these features by moving an on-screen pointer with a mouse or joystick.

Although some people were skeptical that *GEOS* could run fast enough on an eight-bit machine, even the unfinished version at CES ran surprisingly smoothly. By moving the pointer to a menu bar at the top of the screen and pressing the joystick button, you can select various options for managing files and running applications (see photo). To rename a file, for instance, you pull down the File menu and choose Rename. The new filename you enter appears on the directory or beneath the program's icon. To delete a file, you point to its icon, press the joystick button, and "drag" the icon to the trash can icon. To print a file, you drag its icon to the printer icon.

The screen would get cluttered if the icons for every file were displayed at once, so *GEOS* lets you flip through windows as if they were pages in a book. The page number of the current window is displayed near its bottom margin. As the accompanying photo shows, *GEOS* uses the Commodore 64's high-resolution graphics screen to display a smaller-sized character set.



The desktop screen from GEOS, a remarkable Macintosh-like operating system for the Commodore 64. It offers pull-down menus, icons, windows, desk accessories, and custom type fonts, and also speeds up disk access five to seven times.

The desktop includes a full range of Macintosh-like desk accessories, such as a calculator, notepad, alarm clock, photo and text albums (for transferring data between applications), and a preferences window. The preferences window lets you adjust screen colors, change the shape and speed of the pointer, set the date and time, and so forth. The desk accessories can be called up while running other applications—if you're using the word processor and need to perform a calculation, for example, you can pop open the calculator, get your answer, close the calculator, and continue writing.

GEOS comes with two applications: *geoWrite*, a what-you-see-is-what-you-get word processor that lets you type on-screen with several different proportional fonts, and *geoPaint*, a graphics-drawing program with vertical and horizontal scrolling that lets you create images as large as an 8½ × 11-inch page (80 dots per inch resolution).

The price for the entire package—*GEOS*, the desktop and desk accessories, *geoWrite* and *geoPaint*—is $59.95. It was designed by Berkeley Softworks (Berkeley, California) and is scheduled to be available early this spring.

Timeworks (Deerfield, Illinois) is releasing three new programs for the Commodore 128 and one for the 64. *Partner 128* ($59.95) and *Partner 64* ($49.95) are desktop management programs similar to Borland International's *Sidekick* for the IBM PC. Both *Partner* programs include a multifunction business calculator, memo pad, appointment calendar, typewriter, address book, phone book, envelope addresser, and screen printer.

SwiftCalc 128 is a spreadsheet that takes advantage of the 128's 80-column mode. It has programmable keys, multilevel sorting, and several ways of charting data (including 3-D) that can be shown on the screen or printed out. Those who already own the original *SwiftCalc* for the Commodore 64 can upgrade to the 128 version for a nominal fee. Timeworks has also adapted *Sylvia Porter's Personal*

Financial Planner ($69.95) for the 128's 80-column mode.

Cardco (Wichita, Kansas) announced a *Sidekick*-style product, too. Called *Side Saddle* (Cardco is negotiating with Borland for the *Sidekick* name), it offers quick access to a calculator, appointment calendar, telephone directory/ dialer, memo pad, screen printer, and disk functions. It comes on a cartridge for the Commodore 64, with a 128 version to follow. Another interesting Cardco program is *Freeze Frame*, a screen printer that Cardco claims can capture any 64 screen on paper, even with commercial software. It works with any printer that emulates the Commodore 1525, and all Epson- and Okidata-compatibles. A 128 version of *Freeze Frame* is also planned.

The Commodore 128 got another boost when Cardco announced its Personal Productivity Series for the 128's CP/M mode. The first three products in the series are *Personal Accountant*, a financial manager for home or small business; *Personal Inventory*, for figuring your net worth; and *Personal Time Manager*, which can handle up to 26 events for as many as 240 people. Each program sells for $39.95.

In addition to software, Cardco announced three hard disk drives for the 64 and 128. Available in late March for the 64 and shortly afterward for the 128, the drives will come in 5-, 10-, and 20-megabyte capacities for $599.95, $899.95, and $1,299.95, respectively. They're capable of loading a full-function spreadsheet program in 2½ seconds.

Access Software (Salt Lake City, Utah) is releasing a number of products for the Commodore 64, 128, Atari, and Apple. The *Mach 128 Enhancement Package* is a cartridge and disk for the 64 and 128. When the cartridge is plugged in, it senses whether a 1541 or 1571 disk drive is connected and—in the case of the 128—which mode is active. Then it engages or disengages disk speed-up routines as appropriate. Typically there's a fivefold increase in loading speed with either the 1541 or 1571. The cartridge also has a system reset switch. In addition, the *Mach 128* disk includes a disk organizer utility, two machine language monitors, and a program that expands BASIC workspace by 4K. The price is $49.95.

The Development System ($79.95) is a professional macro assembler and text editor for the Commodore 128 (128 or 64 mode) and 64. It includes *Spritemaster*, a utility for creating and animating sprites with machine language programs.

Leader Board, The Pro Golf Simulator is a 3-D game that offers a perspective view of the golf course. It's one of the most detailed sports simulations we've ever seen, with multiple 18-hole courses, handicapping, a wide range of clubs, and numerous other variables. The Commodore 64 version should be available immediately for $39.95. *Inside Story—The Anatomy Learning System* is an educational program with 50 high-resolution graphics screens that let you explore the inner working of the human body. For the Commodore 64, the price is $34.95.

For the Atari 400/800, XL, XE, and Apple II series, Access Software is releasing *Raid Over Moscow* and *Beach-Head II: The Dictator Strikes Back*, popular games previously available for the Commodore 64 and 128. In *Raid Over Moscow*, the Soviets have launched a nuclear attack on North America; your job is to deploy stealth bombers from an orbiting space station to destroy the warheads before they hit. It requires at least 48K RAM and sells for $34.95. *Beach-Head II* is the sequel to *Beach-Head* and features speech synthesis, multiple screens, and the choice of playing another person or the computer. It requires at least 48K RAM and sells for $39.95.

Access has signed an agreement with Multibotics, Inc. (Woods Cross, Utah) to market its line of robotic construction sets. The sets—for youngsters and adults—contain snap-together connectors, gears, shafts, clutches, wheels, electric motors, and other parts that make it possible to build all sorts of motorized contraptions that can be controlled by a personal computer. You can also experiment with digitized speech or temporarily turn a computer into a voltmeter or oscilloscope. Four different Multibot sets are available from $59.95 to $199.95. Interfaces are ready for the Commodore 64 and 128, and Access is working on interfaces for Atari, Apple, Amiga, and IBM computers.

If you've got a good memory, you might recall reading some reviews about a year ago of a Commodore 64 word processor called *SkiWriter*. Although the reviews were good, marketing problems kept the program from appearing on store shelves. Now it's been acquired by a British company, Mastertronic (U.S. offices in Frederick, Maryland). Two changes were made—the built-in telecommunications feature was dropped, and the program is being sold on disk instead of cartridge—but the price has been chopped from $69.95 to $15. There's also a Commodore 128 and Apple II version. At the same time, Mastertronics is introducing two more programs for the Commodore 64: *Busicalc* 3, a spreadsheet, and *Instant Recall*, a filer that stores up to 30,000 characters of data. Both of these products sell for $15, too.

Mindscape (Northbrook, Illinois) is bringing out *The Luscher Profile* (Apple, Commodore 64, IBM PC, and Mac), which constructs a psychological profile based on a person's reactions to colors and shapes; *The American Challenge: A Sailing Simulation* (Apple and IBM), an unusual America's Cup simulation that can be played by two people in remote locations using computers and modems; *Dick Francis' High Stakes* (Apple and IBM), an interactive text adventure that puts you in the role of a wealthy English horse owner; a talking Macintosh version of *Racter*, the AI (artificial insanity) program that holds bizarre conversations with humans; and *Stephen King's The Mist* and *James Bond: A View To A Kill* (Apple, IBM, and Mac), text adventures based on popular thrillers. All of these programs are $39.95, except for *Racter*, which is $44.95.

And finally, if you can spare $39.95, you can now indulge any Rambo fantasies you might have with a Mindscape program called *Rambo: First Blood Part II*. But *Rambo* isn't the shoot-em-up action game you might expect—it's a text adventure. One of its features is a sophisticated parser that lets you communicate in plain English (which is more than the movie character Rambo can do). It runs on the Apple, IBM, and Macintosh. ©

Tug-A-War

Mark Tuttle, Submissions Reviewer

Don't be fooled by the apparent simplicity of this two-player strategy game. It looks easy on the surface, but it's a stiff test of your concentration and ability to think ahead. The original version was written for the Commodore 64. We've added new versions for the Atari 400/800, XL, and XE, Apple II-series computers, Atari 520ST, Amiga, IBM PC/PCjr, and the TI-99/4A. Since the game is based on colors, every version requires a color monitor or TV. The IBM version requires BASICA and a color/graphics adapter for the PC or Cartridge BASIC for the PCjr. The Atari version requires at least 16K of RAM, and the Amiga version requires at least 512K.

Nearly everyone has played tug of war at one time or another. The traditional game pits two players or teams at opposite ends of a rope. At the middle of the rope is a flag, and each side tries to pull the flag into its territory. "Tug-a-War" is based on a similar concept. In this version, the flag is replaced with a round ball shape, and each player tries to maneuver the ball onto his or her side of the screen. Like many two-player games, the difficulty of Tug-a-War depends somewhat on the intelligence of your opponent. But even at the simplest level, you'll find that skill and foresight are essential to success.

Type in and save the appropriate program below. The rules are the same for every version (except Atari 520ST—see special instructions).

Battle Of The Colors

When you run Tug-a-War, two sets of colored boxes appear, one above the other. The lower, longer series of squares is the playing field. Near the middle of the playfield area is a round ball; the outermost boxes at each end of the playfield represent each player's home position. The players alternate turns, each trying to move the ball in their own direction, until it reaches one of the home squares.

So far, so good—but how do you move the ball? It's done not by pulling a rope, but by changing the colors of boxes in the playfield. The color of the square under the ball determines which direction it moves and how far it travels. On any given turn, the ball can move either one or two squares to the left, or one or two squares to the right. At the top of the screen are four boxes that show you which colors are linked to which directions. For instance, the leftmost box shows you which color makes the ball move one square to the left. The next box to the right shows you which color makes it move *two* squares to the left. The second pair of boxes show you which colors make the ball move in the opposite direction, to the right. By changing the color of the box where the ball is currently located, you can make it move toward your home square.

The playfield contains 11 boxes (9 in the TI version, 10 in the Atari ST version). When the game begins, each of these boxes is randomly given one of the four colors shown at the top of the screen. On each turn, you may change the color of one, several, or all of the boxes (however, you must always change at least one box). Below each box is a number which represents its distance from the home position of the player whose turn it is. For instance, if you are the player on the left, then on your turn the boxes are numbered 1, 2, 3, etc., from left to right (the tenth box is marked with a 0, and the eleventh with an A). When it's the right player's turn, the numbering is reversed (the rightmost box is 1, etc.).

To take a turn, you must select a number that corresponds to the numbers shown below the boxes in the playfield. This is done by pressing a single key. Press a number key from 1–0 to select one of the first ten values, or press the A key to choose the eleventh box. The number you choose determines how many boxes change color. For instance, if you press 1, only one box (the one nearest your home square) changes color. If you press 2, the two boxes nearest your home box change, and so on.

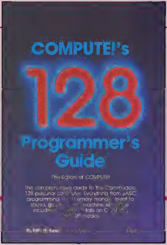
Where do the new colors come from? Every box cycles through the same series of four colors shown in the uppermost set of boxes, going from left to right. For example, if the colors shown there are white-blue-red-purple (the exact colors may be different on your computer), then a white square always changes to blue; a blue square always changes to red; a purple square changes to white, and so on. In other words, the box's current color determines which color it gets after the next color change.

Though every turn involves at least one color change, the ball doesn't necessarily move on every turn. It only moves when you change all the boxes between your home position and the current position of the ball. For example, if the ball is three boxes away from your home square, then you must change the color of at least three boxes in order to move it at all.

Foresight Rewarded

As you can see, there's much more to this game than appears on the surface. At first you might be tempted to try to move the ball as

COMPUTE!'s PROGRAMMER'S GUIDES



Everything you need for successful, entertaining, and challenging programming on your Amiga, Atari ST, or Commodore 128 computer.

Each book is carefully written in COMPUTE!'s lively, understandable style to help even beginning programmers learn the ins and outs of their personal computers.

COMPUTE!'s 128 Programmer's Guide

ISBN 0-87455-031-9
Editors of COMPUTE!   464 pages

Written and compiled by the most technically proficient authors in consumer computing today, the technical staff of COMPUTE! Publications, this guide to the powerful Commodore 128 computer contains a wealth of information for every programmer. Explore BASIC 7.0 through countless hands-on examples and sample programs. Learn how to create dazzling graphics and sophisticated sounds in both BASIC and machine language. See how to program peripherals, such as disk drives, printers, and modems. Enter the world of CP/M, just one of the three modes of the 128. There are even chapters introducing you to machine language programming and the computer's method of managing memory. *COMPUTE!'s 128 Programmer's Guide* includes numerous appendices covering everything from error messages to memory maps.

$16.95

Look for these new books at a bookstore or a computer store near you. Or order directly from COMPUTE! Books. Call toll-free 1-800-346-6767. In NY call 212-887-8525.

COMPUTE!'s ST Programmer's Guide

0-87455-023-8
Editors of COMPUTE!

Complete and comprehensive, yet easy to understand, *COMPUTE!'s ST Programmer's Guide* is a must-buy for any Atari ST owner. The technical staff of COMPUTE! Publications has put together a reference guide to programming that takes the reader through every aspect of this newest Atari personal computer. Logo and BASIC, the two programming languages now available for the machine, are explored in detail. From programming concepts to writing programs, the scores of ready-to-type-in examples show just what can be done, and how to do it. Advanced features of this new-generation computer, such as GEM and TOS, the ST's user interface and operating system, are illustrated as readers write their own applications. Valuable appendices provide information programmers need, including GEM VDI opcodes and a list of ST resources.

$16.95

COMPUTE!'s Amiga Programmer's Guide

0-87455-028-9
Edited

Covering AmigaDOS, BASIC, Intuition, C, machine language, and the other important programming tools which accompany the new Amiga, *COMPUTE!'s Amiga Programmer's Guide* is a clear and thorough guide to the inner workings of this fascinating, new-generation computer. The great speed of its 68000 microprocessor, coupled with the versatility of the Amiga-specific graphics and sound chips, makes the Amiga one of the most powerful computers available today.

$16.95

COMPUTE! Publications,Inc. abc
One of the ABC Publishing Companies

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England often as possible. But that's usually a losing strategy. Remember, the *direction* the ball moves depends on the color of its square before you take the turn.

In many cases, you'll want to move the ball only if it's on a color that moves it toward your goal. But like other games of strategy, Tug-a-War rewards the player who looks beyond the current move and tries to set things up for future moves; sometimes it's wise to make a small, temporary sacrifice in order to benefit later in the game. Because the boxes change colors in the same sequence, the effect of your own move is always completely predictable. However, since a single turn can change the color of many boxes, dramatic changes of fortune are also possible.

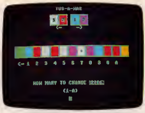
Amiga And 520ST Versions

Since the mouse is an integral system feature on both the Amiga and ST, both of these versions substitute mouse input for keyboard input. To select a square, simply move the mouse pointer to the desired box and press the left mouse button. Because keyboard prompts are unnecessary, no numbers are displayed below the playfield boxes.

Before entering BASIC to load the ST version, you should switch to the low-resolution graphics mode (use the Set Preferences option in the desktop's Options menu). Also, if your ST has 512K and a disk-based operating system, before running the program you should turn off buffered graphics (controlled by the Buffer Grph option in the Settings menu; it's off when no check appears beside the option in the menu). The standard 520ST leaves only about 5K free for BASIC programs, so Tug-a-War won't fit into memory unless the buffered graphics option is turned off. The program fits with buffered graphics switched on only if you have a 1040ST, or a 520ST with memory expansion, or a 520ST that has been upgraded with the TOS operating system in ROM chips (Read Only Memory).

The Amiga version uses the computer's built-in speech feature to announce the players' turns. In other respects, these games work exactly like the others.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.


TUG-A-WAR

<- 1 2 3 4 5 6 7 8 9 0 A

HOW MANY TO CHANGE

(1-A)

"Tug-a-War" for the Commodore 64 and 128 is a game that looks simple, but demands good concentration and foresight.

Program 1: Tug-A-War For Commodore 64/128

```
JJ 100 POKE532B0,0:PRINT"{CLR}
       ";:BC=53281:POKEBC,5:PC
       (1)=5:PC(2)=7:PS=6:CR=1
       :PL=1:X=20
CE 110 B=55715:A$="O{Y}P{DOWN}
       {3 LEFT}{H} {N}{DOWN}
       {3 LEFT}L{P}@":P$(1)="G
       REEN{RVS}{BLK} {OFF}
       {LEFT}":P$(2)="YELLOW":
       B$="{2 UP}"
DM 120 TM$="{HOME}{9 DOWN}":Q$
       ="{RVS}{BLK}{34 SPACES}
       ":DIMCL(11)
QG 130 FORI=0TO23:PRINT"{RVS}
       {BLK}{39 SPACES}"
KE 140 POKE1063+(40*I),160:POK
       E55335+(40*I),0:NEXT
AQ 150 PRINT"{RVS}{39 SPACES}
       {HOME}":POKE2023,160:PO
       KE56295,0
DG 160 PRINT"{HOME}"SPC(15)"
       {RVS}TUG-A-WAR":PRINTSP
       C(13)"{DOWN}{RVS}{WHT}"
       A$B$"{RED}"A$B$"{BLK}
       {CYN}"A$B$"{PUR}"A$B$
RS 170 PRINTSPC(14)"{BLK}1"SPC
       (2)"2"SPC(3)"1"SPC(2)"2
       {DOWN}":PRINTSPC(15)"
       {RVS}<C"SPC(5)"C>"
HK 180 GOSUB620:PRINTTM$"
       {DOWN}{RVS} {GRN}"A$B$;
       :FORZ=1TO11:Y=INT(4*RND
       (1))+1:CL(Z)=Y:POKE646,
       Y
FQ 190 PRINT"{RVS}"A$B$;:NEXTZ
       :PRINT"{YEL}{RVS}"A$
CB 210 POKE646,PEEK(B)AND15:PR
       INTTM$"{2 DOWN}"SPC(X)"
       {RVS}Q":POKEBC,PC(PL)
SF 220 PRINTTM$"{14 DOWN}"SPC(
       18)"{RVS}{BLK}
       {3 SPACES}"
HR 230 AN=0:PRINTTM$SPC(7)"
       {10 DOWN}{BLK}{RVS}HOW
       {SPACE}MANY TO CHANGE
       {OFF}"P$(PL)
MA 240 PRINTSPC(17)"{DOWN}
       {RVS}(1-A){2 DOWN}
       {3 LEFT}{I}{LEFT}";
BR 250 POKE204,0:POKE19B,0:WAI
       T198,1:GETMT$
RB 260 IFASC(MT$)<4BORASC(MT$)
       <>65ANDASC(MT$)>57THEN2
       50
XR 270 POKE204,1:IFMT$="A"THEN
       AN=11:MT$="{LEFT}ALL":G
       OTO300
CR 280 IFMT$="0"THENAN=10:MT$=
       "10":GOTO300
HC 290 AN=VAL(MT$)
KM 300 PRINT"{2 LEFT}
       {5 SPACES}{3 LEFT}"MT$
?FS 310 IFAN<1ORAN>11THEN220
FF 320 IFPL=2THENAN=12-AN:GOTO
       440
HE 330 IFAN<PSTHENCK=1
KQ 340 FORQ=1TOAN:IFCL(Q)=4THE
       NCL(Q)=1:GOTO360
GE 350 CL(Q)=CL(Q)+1
KJ 360 NEXTQ:PRINTTM$:PRINT"
       {BLK}{RVS} {GRN}"A$B$;:
       FORZ=1TO11:POKE646,CL(Z
       ):PRINT"{RVS}"A$B$;
QD 370 NEXTZ:PRINT"{YEL}{RVS}"
       A$:POKE646,PEEK(B)AND15
       :PRINT"{HOME}{11 DOWN}"
       SPC(X)"{RVS} "
FH 3B0 IFCK=1THENCK=0:GOTO400
MH 390 ONPEEK(B)AND15GOSUB490,
       500,510,520
BF 400 IFPS<1THENPL=1:WC=5:B=5
       569B:X=2:GOTO530
DP 410 IFPS>11THENPL=2:WC=7:B=
       55734:X=3B:GOTO530
KD 420 IFPL=1THENPL=2:GOSUB640
       :GOSUB630:GOTO210
HE 430 PL=1:GOSUB640:PRINT"
       {10 UP}":GOSUB620:GOTO2
       10
XH 440 FORQ=ANTO11
DH 450 IFAN<1ORAN>11THEN220
BD 460 IFAN>PSTHENCK=1
GJ 470 IFCL(Q)=4THENCL(Q)=1:GO
       TO360
HB 480 CL(Q)=CL(Q)+1:GOTO360
JX 490 B=B+6:X=X+6:PS=PS+2:RET
       URN
PA 500 B=B-3:X=X-3:PS=PS-1:RET
       URN
XM 510 B=B-6:X=X-6:PS=PS-2:RET
       URN
BE 520 B=B+3:X=X+3:PS=PS+1:RET
       URN
FF 530 POKE646,PEEK(B)AND15:PR
       INT"{HOME}{11 DOWN}"SPC
       (X)"{RVS}Q":PRINTTM$"
       {10 DOWN}";
PG 540 FORE=1TO5:PRINTQ$:NEXT:
       GOSUB640:GOSUB640
FM 550 PRINT"{HOME}{11 DOWN}"S
       PC(11)P$(PL)" IS THE WI
       NNER":Z=WC:FORI=1TO11:P
       OKEBC,Z
CP 560 IFZ=0THENZ=WC:GOSUB610:
       NEXT
SJ 570 Z=0:GOSUB610:NEXT
BP 580 POKEBC,15:PRINTTM$SPC(1
       1)"{10 DOWN}{RVS}LIKE T
       O PLAY AGAIN{2 DOWN}
       {11 LEFT}{RVS}Y/N"
RA 590 POKE19B,0:WAIT19B,1:GET
       MT$:IFMT$<>"N"THENRUN
JC 600 POKE198,0:SYS198
QC 610 FORP=1TO200:NEXTP:RETUR
       N
CG 620 PRINTTM$"{5 DOWN}{RVS}
       {BLK}{2 SPACES}<C 1
       {2 SPACES}2{2 SPACES}3
       {2 SPACES}4{2 SPACES}5
       {2 SPACES}6{2 SPACES}7
       {2 SPACES}B{2 SPACES}9
       {2 SPACES}0{2 SPACES}A"
       :RETURN
AA 630 PRINTTM$"{5 DOWN}{RVS}
       {BLK}{5 SPACES}A
       {2 SPACES}0{2 SPACES}9
       {2 SPACES}8{2 SPACES}7
       {2 SPACES}6{2 SPACES}5
```

From the publishers of *COMPUTE!*

Programming Languages Communicating With You Compute!
COMPUTE!
The Leading Magazine Of Home, Educational, And Recreational Computing
Amiga's
Amazing Graph
Telecommunications Software Overview
72 Inexpensive Packages For Commodore, Atari, Apple, Macintosh, IBM
Commodore 64 3-D Animated Graphics
Simple Commands To Create And Move Your Own Shapes
Apple Program Protector
Shield Your Programs From Prying Eyes
Atari REMover
Automatically Delete REMs From BASIC
RE
COMPUTE! DISK

April 1986 *COMPUTE!* Disk

All the exciting programs from the past three issues of *COMPUTE!* are on one timesaving, error-free floppy disk that is ready to load on your Commodore 64 and 128 computers. The April 1986 *COMPUTE! Disk* contains the entertaining and useful Commodore 64 and 128 programs from the February, March, and April 1986 issues of *COMPUTE!*.

The April 1986 *COMPUTE! Disk* costs $12.95 plus $2.00 shipping and handling and is available only from COMPUTE! Publications.

For added savings and convenience, you may also subscribe to the *COMPUTE! Disk.* At a cost of only $39.95 a year (a $12.00 savings), you'll receive four disks, one every three months. Each disk will contain all the programs for your Commodore machine from the previous three issues of *COMPUTE!*.

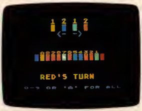
This is an excellent way to build your software library while you enjoy the quality programs from *COMPUTE!*.

Disks and subscriptions are available for Apple, Atari, Commodore 64 and 128, and IBM personal computers. Call for details.

For more information or to order the February 1986 *COMPUTE! Disk,* call toll free 1-800-346-6767 (in NY 212-265-8360) or write *COMPUTE! Disk,* P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE! Publications, Inc. abc
One of the ABC Publishing Companies

```
          {2 SPACES}4{2 SPACES}3
          {2 SPACES}2{2 SPACES}1
          {SPACE}C>":RETURN
AQ 640 PRINTTM$"{5 DOWN}{RVS}
          {BLK}{39 SPACES}":RETUR
          N
```

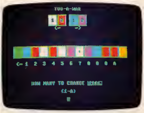


This version of "Tug-a-War" runs on all Atari 400, 800, XL, and XE computers.

Program 2: Tug-A-War For Atari 400/800, XL, XE

Version by Kevin Mykytyn, Editorial Programmer

```
OC 10 CB=PEEK(106)-B:POKE 10
      6,CB-4:GRAPHICS 18:DIM
       C$(4),COL(11),MOV(4),
      K$(1):C$="Z▨z▨"
BI 20 MOV(1)=-1:MOV(2)=-2:MO
      V(3)=1:MOV(4)=2:OPEN #
      1,4,B,"K:":GOSUB 190:G
      OSUB 250
KL 30 BP=6:PL=0
AP 40 PL=(PL=0):GOSUB 290:PO
      KE 5324B,72+B*BP
NK 50 GET #1,K:K$=CHR$(K):IF
       K$="A" THEN SP=11:GOT
      O 90
GB 60 IF K$="0" THEN SP=10:G
      OTO 90
HF 70 IF K$<"1" OR K$>"9" TH
      EN 50
OL B0 SP=VAL(K$)
CK 90 IF (PL=0 AND 12-SP>BP)
       OR (PL=1 AND SP<BP) T
      HEN 120
HO 100 BP=BP+MOV(COL(BP)):IF
        BP>12 THEN BP=12
HM 110 IF BP=-1 THEN BP=0
KI 120 GOSUB 320
LK 130 IF BP>0 AND BP<12 THE
       N 40
MG 140 POKE 5324B,72+B*BP:PO
       SITION 4,9
EN 150 IF BP=0 THEN PRINT #6
       ;" BLUE WINS! ":GOTO
       170
DJ 160 PRINT #6;" RED WINS!
       "
CI 170 POSITION 0,11:PRINT #
       6;"{3 SPACES}press an
       y key{3 SPACES}"
BK 1B0 GET #1,K:GOTO 30
JH 190 NS=CB*256:POSITION 5,
       5:PRINT #6;"PLEASE WA
       IT"
NF 200 FOR A=0 TO 1023:POKE
       NS+A,PEEK(57344+A):NE
       XT A
FC 210 POKE 756,CB:FOR A=NS+
       464 TO NS+471:POKE A,
       126:NEXT A
CO 220 FOR A=NS+1536 TO NS+1
       920:POKE A,0:NEXT A:F
       OR A=NS+1596 TO NS+16
       03:POKE A+256,126:POK
       E A+128,126:NEXT A
BI 230 POKE 559,46:POKE 623,
       1:POKE 53250,72:POKE
       53249,16B:POKE 53277,
       3:POKE 54279,CB+4:POK
       E 704,15:POKE 705,52
JB 240 FOR A=NS+159B TO NS+1
       601:READ B:POKE A,B:N
       EXT A:POKE 706,132:RE
       TURN :DATA 24,60,60,2
       4
FP 250 PRINT #6;"{CLEAR}":PO
       SITION 6,0:PRINT #6;"
       1 2 1 2":POSITION 6,1
       :PRINT #6;"Z ▨ z ▨":P
       OSITION 7,2:PRINT #6;
       "◀ ▶"
DJ 260 GB=PEEK(560)+256*PEEK
       (561):POKE GB+10,6:PO
       KE GB+16,6
KL 270 POSITION 4,6:FOR A=1
       TO 11:Q=INT(RND(1)*4+
       1):COL(A)=Q:PRINT #6;
       C$(Q,Q);:NEXT A
IM 2B0 POSITION 0,11:PRINT #
       6;" 0-9 OR 'A' FOR AL
       L":RETURN
KJ 290 POSITION 4,9
FG 300 IF PL=1 THEN PRINT #6
       ;"BLUE'S TURN":POSITI
       ON 4,5:FOR A=1 TO 9:P
       RINT #6;A;:NEXT A:PRI
       NT #6;"0A":RETURN
BH 310 PRINT #6;"RED'S TURN
       ":POSITION 4,5:PRINT
       #6;"A0";:FOR A=9 TO 1
        STEP -1:PRINT #6;A;:
       NEXT A:RETURN
PD 320 IF PL=0 THEN 340
JI 330 FOR A=1 TO SP:GOSUB 3
       50:NEXT A:RETURN
PE 340 FOR A=11 TO 12-SP STE
       P -1:GOSUB 350:NEXT A
       :RETURN
LH 350 COL(A)=COL(A)+1-4*(CO
       L(A)=4):POSITION 3+A,
       6:PRINT #6;C$(COL(A),
       COL(A)):RETURN
```



"Tug-a-War" for the IBM PC and PCjr.

Program 3: Tug-A-War For IBM PC/PCjr

Version by Kevin Mykytyn, Editorial Programmer

```
GE 10 GOTO 30
GN 20 FOR ROW=0 TO 2:LOCATE Y+RO
      W,PS*3+X:PRINT B$:NEXT:RET
      URN
NO 30 KEY OFF:SCREEN 0,0:WIDTH 4
      0:B$=CHR$(222)+CHR$(219)+C
      HR$(221)+CHR$(31)
HP 40 DIM COL(11):BP=6:C(1)=2:C(
      2)=6:C(3)=3:C(4)=5:PL=0:RA
      NDOMIZE TIMER
LA 50 MOV(1)=-1:MOV(2)=-2:MOV(3)
      =1:MOV(4)=2
LP 60 GOSUB 190
CI 70 PL=(PL=0):GOSUB 230:GOSUB
      240:GOSUB 1B0
LA B0 K$=INKEY$:IF K$="a" OR K$=
      "A" THEN SP=11:GOTO 110
OJ 90 IF K$="0" THEN SP=10:GOTO
      110
HE 100 IF K$<"1" OR K$>"9" THEN
       B0 ELSE SP=VAL(K$)
HG 110 IF (PL=0 AND 12-SP>BP) OR
        (PL=-1 AND SP<BP) THEN 1
       30
NL 120 BP=BP+MOV(COL(BP)):IF BP>
       12 THEN BP=12 ELSE IF BP=
       -1 THEN BP=0
NO 130 GOSUB 260:IF BP>0 AND BP<
       12 THEN 70
LJ 140 GOSUB 230:LOCATE 20,15:CO
       LOR 9:IF BP=0 THEN PRINT
       " Blue Wins! ":GOTO 160
GL 150 COLOR 4:PRINT " Red Wins!
        "
FA 160 LOCATE 23,B:COLOR 14:PRIN
       T "Press any key to play
       again":GOSUB 1B0
GJ 170 A$=INKEY$:IF A$="" THEN 1
       70 ELSE RUN
FF 1B0 DEF SEG=0:POKE 1050,PEEK(
       1052):RETURN
BE 190 CLS:Y=3:X=0:FOR A=1 TO 4:
       PS=4+A:COLOR C(A):GOSUB 2
       0:NEXT
FE 200 COLOR 14:PRINT SPC(15)CHR
       $(17)" "STRING$(2,17)"  "
       CHR$(16)"  "STRING$(2,16)
EI 210 Y=13:X=1:COLOR 9:PS=0:GOS
       UB 20:FOR A=1 TO 11:Q=INT
       (RND(1)*4+1):COL(A)=Q:COL
       OR C(Q):PS=A:GOSUB 20:NEX
       T:COLOR 4:PS=12:GOSUB 20
IL 220 LOCATE 23,B,0:COLOR 10:PR
       INT "Press (0-9) or 'A' f
       or all":RETURN
GF 230 COLOR 15:LOCATE 14,BP*3+2
       :PRINT CHR$(219):RETURN
BG 240 LOCATE 20,15:IF PL=0 THEN
        COLOR 4:PRINT "Red's Tur
       n ":LOCATE 11,5:PRINT "A
        0 ";:FOR A=9 TO 1 STEP-1
       :PRINT A;:NEXT:RETURN
LF 250 COLOR 9:PRINT "Blue's Tur
       n":LOCATE 11,4:FOR A=1 TO
        9:PRINT A;:NEXT:PRINT "
       0  A":RETURN
HI 260 IF PL=0 THEN 280
BO 270 FOR A=1 TO SP:GOSUB 290:N
       EXT:RETURN
FP 2B0 FOR A=11 TO 12-SP STEP-1:
       GOSUB 290:NEXT:RETURN
DE 290 COL(A)=COL(A)+1+4*(COL(A)
       =4):COLOR C(COL(A)):PS=A:
       GOSUB 20:RETURN
```

Program 4: Tug-A-War For Apple

Version by Tim Victor, Editorial Programmer

```
3B 100 GOSUB 400
BA 110 HGR : HOME
5C 120 HCOLOR= 3: FOR TD = - 1 T
       O 1 STEP 2: FOR TN = 0 TO
        1: FOR TX = - 1 TO TN ST
       EP 2: GOSUB 500: NEXT : N
       EXT : NEXT
DC 130 VP = 40
A1 140 FOR I = 0 TO 3:HC = CT(I)
       :PS = 4.5 + I: GOSUB 430:
        NEXT
EB 150 VP = 146:HC = 1:PS = 0: G
       OSUB 430: GOSUB 460
AE 160 FOR I = 0 TO 10:BC(I) = I
       NT ( RND (1) * 4):HC = CT
       (BC(I)):PS = I + 1: GOSUB
        430: NEXT
```

From the publishers of *COMPUTE!*

April 1986 *COMPUTE!* Disk

All the exciting programs from the past three issues of *COMPUTE!* are on one timesaving, error-free floppy disk that is ready to load on your IBM PC and PCjr computers. The April 1986 *COMPUTE! Disk* contains the entertaining and useful IBM programs from the February, March, and April 1986 issues of *COMPUTE!*.

The April 1986 *COMPUTE! Disk* costs $12.95 plus $2.00 shipping and handling and is available only from COMPUTE! Publications.

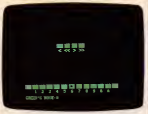
For added savings and convenience, you may also subscribe to the *COMPUTE! Disk*. At a cost of only $39.95 a year (a $12.00 savings), you'll receive four disks, one every three months. Each disk will contain all the programs for your IBM machine from the previous three issues of *COMPUTE!*.

This is an excellent way to build your software library while you enjoy the quality programs from *COMPUTE!*.

Disks and subscriptions are available for Apple, Atari, Commodore 64 and 128, and IBM personal computers. Call for details.

For more information or to order the February 1986 *COMPUTE! Disk*, call toll free 1-800-346-6767 (in NY 212-265-8360) or write *COMPUTE! Disk*, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

COMPUTE! Publications,Inc. abc
One of the ABC Publishing Companies



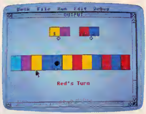
Apple "Tug-a-War," a challenging strategy game.

```
C9 170 HC = 6:PS = 12: GOSUB 430
       : GOSUB 460
F9 180 BP = 5: GOSUB 470
69 190 VTAB 21: FOR I = 1 TO 11:
        HTAB I * 3 + 2: IF I < 1
       Ø THEN PRINT CHR$ (48 + I
       );
89 200 IF I = 1Ø THEN PRINT "Ø";
15 210 IF I = 11 THEN PRINT "A";
E4 220 NEXT : VTAB 23: HTAB 1: P
       RINT "GREEN'S MOVE:";
46 230 GOSUB 52Ø:A = A - 1: IF (
       BP < = A) THEN BP = BP +
       JT(BC(BP))
97 240 FOR I = Ø TO A:BC(I) = BC
       (I) + 1 - 4 * (BC(I) = 3)
       :HC = CT(BC(I)):PS = I +
       1: GOSUB 43Ø: NEXT
54 250 GOSUB 47Ø
16 260 IF BP < Ø OR BP > 1Ø THEN
        36Ø
67 270 VTAB 21: FOR I = 1 TO 11:
        HTAB (12 - I) * 3 + 2: I
       F I < 1Ø THEN PRINT CHR$
       (48 + I);
99 280 IF I = 1Ø THEN PRINT "Ø";
25 290 IF I = 11 THEN PRINT "A";
8B 300 NEXT : VTAB 23: HTAB 1: P
       RINT "BLUE'S MOVE: ";
1F 310 GOSUB 52Ø:A = 11 - A: IF
       (BP > = A) THEN BP = BP +
        JT(BC(BP))
5A 320 FOR I = 1Ø TO A STEP - 1:
       BC(I) = BC(I) + 1 - 4 * (
       BC(I) = 3):HC = CT(BC(I))
       :PS = I + 1: GOSUB 43Ø: N
       EXT
51 330 GOSUB 47Ø
13 340 IF BP < Ø OR BP > 1Ø THEN
        36Ø
9F 350 GOTO 19Ø
92 360 PS = 12 * (BP > Ø) - 1: H
       COLOR= 4 * (BP > Ø): BOSU
       B 49Ø
33 370 VTAB 23: HTAB 1: IF BP <
       Ø THEN PRINT "GREEN WINS
         ": GOTO 39Ø
6C 380 IF BP > 1Ø THEN PRINT "BL
       UE WINS    "
8A 390 GET A$: GOTO 11Ø
DE 400 FOR I = Ø TO 3: READ CT(I
       ): NEXT
38 410 FOR I = Ø TO 3: READ JT(I
       ): NEXT : RETURN
84 420 DATA 3,5,6,2,-1,-2,1,2
8C 430 HCOLOR= HC: FOR YP = VP T
       O VP + 1Ø
9A 440 HPLOT PS * 21 + 1,YP TO P
       S * 21 + 17,YP: NEXT
1E 450 RETURN
AB 460 HCOLOR= 3: FOR YP = VP +
       1 TO VP + 9 STEP 2: HPLOT
        PS * 21 + 1,YP TO PS * 2
       1 + 17,YP: NEXT : RETURN
52 470 IF BP < Ø OR BP > 1Ø THEN
        RETURN
19 480 HCOLOR= 4 * (CT(BC(BP)) >
        3):PS = BP
74 490 FOR YP = VP + 3 TO VP + 7
       : HPLOT PS * 21 + 27,YP T
       O PS * 21 + 32,YP: NEXT :
        RETURN
DF 500 TP = 124 + (TD + TN) * 21
        + TN * TX * 4:TL = TP +
       TD * 3:TR = TP - TD * 3
E5 510 HPLOT TR,6Ø TO TL,57 TO T
       R,54: RETURN
86 520 POKE 49168,Ø: GET A$: IF
       A$ = CHR$ (3) THEN END
89 530 IF A$ = CHR$ (3) THEN END
CB 540 IF A$ < > "A" AND A$ < >
       "a" AND (A$ < "Ø" OR A$ >
        "9") THEN 52Ø
36 550 IF A$ = "A" OR A$ = "a" T
       HEN A$ = CHR$ (59)
38 560 IF A$ = "Ø" THEN A$ = CHR
       $ (58)
51 570 A = ASC (A$) - 48: RETURN
```

Program 5: Tug-A-War For Atari ST

Version by Kevin Mykytyn, Editorial Programmer

```
10 fullw 2:clearw 2:color 1,1,1
20 bp=6:c(1)=6:c(2)=7:c(3)=10:c(4)=12:pl
   =0
30 mov(1)=-1:mov(2)=-2:mov(3)=1:mov(
   4)=2
40 gosub 170:gosub 270
50 pl=(pl=0):gosub drawball:gosub play
   er
60 gosub readmouse:if y<98 or y>127 o
   r x<34 or x>273 then 60
70 sp=int((x-11)/24)
80 if (pl=0 and sp>bp) or (pl=-1 and sp
   <bp) then 110
90 t=c(col(bp)):color 1,t,t:pcircle bp*24+2
   2,91,6
100 bp=bp+mov(col(bp)):if bp>11 the
    n bp=11 else if bp=-1 then bp=0
110 gosub colchange
120 gosub drawball:if bp>0 and bp<11 th
    en 50
130 gotoxy 13,14:if bp=0 then color 5:pri
    nt " Blue Wins! ":goto 150
140 color 2:print " Red Wins! "
150 gotoxy 10,16:color 1:print "Press Mou
    se Button":gosub readmouse:clear: go
    to 10
160 drawball: color 1,1,1:pcircle bp*24+2
    2,91,6:return
170 for a=75 to 105 step 30:linef 10,a,298,
    a:next
180 for a=10 to 298 step 24:linef a,75,a,10
    5:next
190 color 1,5:fill 12,77:color 1,2:fill 296,77
200 gotoxy 12,3:print "1 2 1 2"
210 for a=100 to 220 step 24:linef a,19,a,3
    7:next
220 for a=19 to 37 step 18:linef 100,a,148,
    a:linef 172,a,220,a:next
230 color 1,6:fill 101,20:color 1,7:fill 125,20
240 color 1,10:fill 173,20:color 1,12:fill 197,
    20
250 gotoxy 13,4:print chr$(4);" ";chr$(3)
260 return
270 for a=1 to 10:q=int(rnd(1)*4+1):col(a
    )=q:color 1,c(q):fill 25+a*24,77
280 next:return
290 readmouse: poke contrl,124
300 poke contrl+2,0:poke contrl+6,0
310 vdisys(0):if peek(intout)=0 then 310
320 x=peek(ptsout):y=peek(ptsout+2)
330 return
340 colchange: if pl=0 then 360
350 for a=1 to sp:gosub 370:next:return
360 for a=10 to sp step-1:gosub 370:next:r
    eturn
370 col(a)=col(a)+1+4*(col(a)=4)
380 color 1,c(col(a)):fill 25+a*24,77
390 return
400 player: gotoxy 13,14:if pl=0 then colo
    r 2:print "Red's Turn ":return
410 color 5:print "Blue's Turn":return
```



Use the mouse to play the Atari ST version of "Tug-a-War."

Program 6: Tug-A-War For TI-99/4A

Version by Patrick Parrish, Programming Supervisor

```
100 GOTO 150
110 FOR I=1 TO LEN(A$)
120 CALL HCHAR(R,C+I,ASC(SE
    G$(A$,I,1)))
130 NEXT I
140 RETURN
150 RANDOMIZE
160 CALL COLOR(14,1,7)
170 CALL SCREEN(2)
180 PC(Ø)=5
190 PC(1)=7
200 P$(Ø)="BLUE"
210 P$(1)="RED"
220 Y$(Ø)="<- 1  2  3  4  5
      6  7  8  9   "
230 Y$(1)="    9  8  7  6  5
      4  3  2  1 ->"
240 KHAR(Ø)=Ø
250 KHAR(1Ø)=5
260 FOR I=96 TO 136 STEP 8
270 CALL CHAR(I,"ØØØØØØØØØØ
    ØØØØØØ")
280 CALL CHAR(I+1,"ØFØFØFØF
    ØFØFØFØF")
290 CALL CHAR(I+2,"3Ø78FCFC
    FC7830")
300 CALL CHAR(I+3,"ØØ1Ø3Ø1Ø
    1Ø1Ø1Ø38")
310 CALL CHAR(I+4,"ØØ384404
    Ø81Ø2Ø7C")
320 NEXT I
330 PS=5
340 PL=Ø
350 BP=17
360 CALL CLEAR
370 GOSUB 1ØØØ
380 PRINT TAB(11);"TUG-A-WA
    R"
390 PRINT : :
400 B$=CHR$(128)&CHR$(128)&
    CHR$(129)
410 PRINT TAB(9);"hhippq xx
    y";B$
420 PRINT TAB(9);"hkiptq x{
    y";CHR$(128);CHR$(132);
    CHR$(129)
430 PRINT TAB(9);"hhippq xx
    y";B$
```

COMPUTE!'s *All New* Apple Applications Special

COMPUTE!'s latest *Apple Applications Special* features in-depth articles and interviews, all the inside news about Apple, clearly written tutorials, software buyer's guides, new product information, and valuable ready-to-type-in programs for all Apple users.

Inside A MacArtist
A Tutorial Feature On The Inner Workings Of MacPaint, MacDraw, And MacWrite
COMPUTE!'s
Apple
SPECIAL:
APPLICATIONS ISSUE
$3.95
Vol 1 No 1
Business Applications Software Buyer's Guide
SALES
Getting Ready For Telecommunications
Exploring Databases
Jazz
A First Look At Jazz For The Macintosh
ABC
Apple In Education
A Game-Lover's Choices: The Ten Best-Ever Apple Games
Ready-To-Use Programs Free With This Issue:
Heat Seeker: A fast-paced arcade style game
Chess: Five levels of challenging play await you
Spelling Bee: A word-guessing game for youngsters
Softsearcher: A machine language search routine
And more
APPLE II DISK Available See page 11

Apple owners find these special Apple issues the most understandable, complete, and valuable resources available today.

PLUS

All the programs in *COMPUTE!'s Apple Applications Special* are also available on a timesaving disk, ready to run on your Apple II, II+, IIe, and IIc. The *Disk* costs only $12.95 (plus $2.00 shipping and handling) and gives you immediate access to all the great programs in this special issue.

Look for the Spring/Summer 1986 issue of *COMPUTE!'s Apple Applications Special* on sale where you buy other COMPUTE! publications, or order directly from COMPUTE!. This special issue goes on sale April 8, 1986.

Send in the attached order card or call toll free 800-346-6767 (in NY call 212-887-8525).

Features

- **Business Applications**

'86 Apple: An Interview with John Scully

A wide-ranging interview with the president of Apple. The company's plans for the coming year, its markets, the new Macintosh, and the viability of the Apple II.

Business Software Buyer's Guide

A buyer's guide to the newest Apple II and Macintosh word processors, databases, spreadsheets, and more.

The Expanding Mac

- **Education**

Apple Rules the Schools

Why does Apple have a lock on educational computing? Comments from teachers, administrators, and Apple.

Computers and the Humanities

Educational Software Buyer's Guide

- **The Expanding Apple**

It's New II

A multitude of new hardware and software for the Apple II line—from color printers to Mac-like software—is evaluated.

Weirdware: Off the Beaten Software Path

Weirdware—out of the ordinary software—can turn the Apple II or Macintosh into a telescope, astrological fortune-teller, baby evaluator, and much, much more.

MacAdds: More for the Macintosh

Applications

- **Utilities and Tutorials**

Windows

Create Macintosh-style windows on any Apple II-series computer. Set window size, open, close, and retrieve information.

Mouse Cursor

A Macintosh BASIC utility for altering the mouse pointer. Design data can be saved, then used in other BASIC programs.

Your Personal Ledger

A complete personal financial application for tracking expenses, income, and assets. Easy to use, and packed with features from report generation to customized category codes.

Personal Publishing With Your Macintosh

Tutorial and guide to using such software as *MacPaint* and *MacWrite* to customize letterheads, cards, banners, and more.

Keynote

- **Education and Recreation**

Lexitron

Entertaining word game where players try to beat the clock while finding as many hidden words as possible.

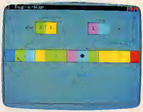
Backgammon

Play the computer in this classic game. This version observes all the rules of standard backgammon.

Apple Automatic Proofreader

COMPUTE!™ Publications,Inc. abc
Part of ABC Consumer Magazines, Inc
One of the ABC Publishing Companies
825 7th Avenue, 6th Floor, New York, NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk COMPUTE! Books and COMPUTE!'s Apple Applications

```
440 PRINT
450 PRINT TAB(11);"<-";TAB(
    18);"->"
460 FOR I=1 TO 15
470 PRINT
480 NEXT I
490 FOR I=1 TO 9
500 RANDOMIZE
510 KHAR(I)=INT(4*RND)+1
520 NEXT I
530 FOR R=13 TO 15
540 CALL HCHAR(R,2,96,2)
550 FOR I=1 TO 9
560 KH=96+KHAR(I)*8
570 CALL HCHAR(R,I*3+1,KH)
580 CALL HCHAR(R,I*3+2,KH)
590 CALL HCHAR(R,I*3+3,KH+1
    )
600 NEXT I
610 CALL HCHAR(R,31,136,2)
620 NEXT R
630 CALL HCHAR(14,BP,96+KHA
    R(PS)*8+2)
640 IF (PS=0)+(PS=10)THEN 1
    180
650 A$=Y$(PL)
660 R=17
670 C=1
680 GOSUB 110
690 CALL HCHAR(24,17,32)
700 A$=P$(PL)&"'S TURN "
710 R=20
720 C=11
730 GOSUB 110
740 R=22
750 C=14
760 A$="(1-9)"
770 GOSUB 110
780 GOSUB 1020
790 CALL KEY(0,K,H)
800 IF H=0 THEN 790
810 IF (K<49)+(K>57)THEN 79
    0
820 AN=K-48
830 CALL HCHAR(24,17,K)
840 IF PL=0 THEN 890
850 AN=10-AN
860 S=AN
870 E=9
880 GOTO 910
890 S=1
900 E=AN
910 GOSUB 1100
920 FOR Q=S TO E
930 IF KHAR(Q)<>4 THEN 960
940 KHAR(Q)=1
950 GOTO 970
960 KHAR(Q)=KHAR(Q)+1
970 NEXT Q
980 PL=-(PL=0)
990 GOTO 530
1000 CALL COLOR(9,1,5)
1010 CALL COLOR(14,1,7)
1020 FOR I=1 TO 8
1030 CALL COLOR(I,PC(PL),2)
1040 NEXT I
1050 CALL COLOR(10,PC(PL),1
     6)
1060 CALL COLOR(11,PC(PL),1
     1)
1070 CALL CØLOR(12,PC(PL),8
     )
1080 CALL COLOR(13,PC(PL),1
     4)
1090 RETURN
1100 IF ((AN<PS)*(PL=0))+(A
     N>PS)*(PL=1)THEN 1170
1110 A=(KHAR(PS)=1)+(KHAR(P
     S)=2)*2-(KHAR(PS)=3)-(
     KHAR(PS)=4)*2
1120 BP=BP+A*3
1130 PS=PS+A
1140 IF (PS>0)*(PS<10)THEN
     1170
1150 PS=-(PS=-1)+(PS=11)+PS
1160 BP=-(BP<5)*3-(BP>29)*3
     1
1170 RETURN
1180 R=14
1190 C=7
1200 A$=P$(-(PS=10))&" IS T
     HE WINNER!"
1210 GOSUB 110
1220 A$="LIKE TO PLAY AGAIN
      (Y/N)?"
1230 R=24
1240 C=4
1250 GOSUB 110
1260 CALL KEY(0,K,H)
1270 IF H=0 THEN 1260
1280 IF (K<>78)*(K<>89)THEN
      1260
1290 IF K=89 THEN 330
```



This version of "Tug-a-War" uses several of the Amiga's 4,096 different color shades.

Program 7: Tug-A-War For Amiga

Version by John Krause, Assistant Technical Editor

```
SAY TRANSLATE$("")←
SCREEN 2,320,200,3,1←
WINDOW 2," Tug-A-War ",,12,2←
FOR i=0 TO 7←
 READ r,g,b←
 PALETTE i,r,g,b←
NEXT←
RANDOMIZE TIMER←
DIM a(11)←
FOR i=1 TO 11←
 a(i)=INT(RND(1)*4)+4←
NEXT←
row=3←
col=3:colr=4:GOSUB frame:GOSUB squ
ars←
col=4:colr=5:GOSUB frame:GOSUB squ
ars←
col=8:colr=6:GOSUB frame:GOSUB squ
are←
col=9:colr=7:GOSUB frame:GOSUB squ
ars←
row=10←
LOCATE 5,11:COLOR 1,4:PRINT "2"←
LOCATE 5,14:COLOR 1,5:PRINT "1"←
LOCATE 5,26:COLOR 1,6:PRINT "1"←
LOCATE 5,29:COLOR 1,7:PRINT "2"←
LINE (64,35)-STEP(-18,0),1←
LINE -STEP(5,4),1←
LINE (46,35)-STEP(5,-4),1←
LINE (248,35)-STEP(16,0),1←
LINE -STEP(-5,4),1←
LINE (264,35)-STEP(-5,-4),1←
FOR col=0 TO 12←
 GOSUB frame←
NEXT←
col=0:colr=3:GOSUB square←
col=12:colr=2:GOSUB square←
dot=6:GOSUB update←
SAY TRANSLATE$("welcome to tug o wa
r ")←
main:←
LOCATE 17,15←
IF red THEN←
 COLOR 2,0:PRINT "Red's turn "←
 SAY TRANSLATE$("reds turn.")←
ELSE←
 COLOR 3,0:PRINT "Blue's turn"←
 SAY TRANSLATE$("blues turn.")←
END IF←
WHILE MOUSE(0)<>1 OR MOUSE(4)<8
0 OR MOUSE(4)>104 OR MOUSE(3)<2
3 OR MOUSE(3)>278←
WEND←
click=INT(MOUSE(3)/24)←
IF (red AND click<=dot) OR (red=0 AN
D click>=dot) THEN ←
 temp=dot←
 IF a(temp)=4 THEN dot=dot-2←
 IF a(temp)=5 THEN dot=dot-1←
 IF a(temp)=6 THEN dot=dot+1←
 IF a(temp)=7 THEN dot=dot+2←
END IF←
IF red THEN←
 FOR i=click TO 11←
  a(i)=a(i)+1←
  IF a(i)=8 THEN a(i)=4←
  NEXT←
ELSE←
 FOR i=1 TO click←
  a(i)=a(i)+1←
  IF a(i)=8 THEN a(i)=4←
  NEXT←
END IF←
IF dot>11 THEN←
 dot=12:GOSUB update←
 LOCATE 17,15:COLOR 2,0:PRINT " Re
d wins! "←
 SAY TRANSLATE$("red wins.")←
 GOTO quit←
END IF←
IF dot<1 THEN←
 dot=0:GOSUB update←
 LOCATE 17,15:COLOR 3,0:PRINT "Blu
e wins! "←
 SAY TRANSLATE$("blue wins.")←
 GOTO quit←
END IF←
GOSUB update←
red=1-red←
GOTO main←
frame:←
x=24:IF 24*col>280 THEN x=23←
LINE (24*col,8*row)-STEP(x,24),1,b←
RETURN←
square:←
x=22:IF 24*col+1>280 THEN x=21←
LINE (24*col+1,8*row+1)-STEP(x,22),c
olr,bf←
RETURN←
update:←
FOR col=1 TO 11←
 colr=a(col):GOSUB square←
NEXT←
CIRCLE (24*dot+11,91),5,1←
PAINT (24*dot+11,91),1←
RETURN ←
quit:←
LOCATE 19,7:COLOR 1,0:PRINT "Clic
k mouse to play again."←
SAY TRANSLATE$("click mouse to pla
y again.")←
WHILE MOUSE(0)=0:WEND←
RUN←
DATA .5,.5,.5,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
0,1,0,1,0,1,1←
```

©

Silent Service

Neil Randall

Requirements: Commodore 64 or 128 (in 64 mode); Apple II-series computer with at least 64K RAM; Atari 400/800, XL, or XE with at least 48K RAM; IBM PC with color/graphics adapter; or an IBM PCjr. A disk drive is also required, and a joystick is recommended. The Commodore version was reviewed.

Silent Service, from Microprose Software, is one of a new type of computer war game. A cross between arcade action games and traditional strategy war games, these new games put you on the battlefield in command of a plane, tank, or submarine. Microprose's *F-15 Strike Eagle* placed you in a modern jet fighter. In *Silent Service*, you're the captain of a U.S. submarine in the Pacific during World War II. Your mission is to sink Japanese cargo, troop, and oil shipping. The game includes several scenarios based on actual engagements.

Silent Service employs several graphics screens to relay the information needed to command the sub. The Patrol Navigation Map shows a 150,000 square-mile area of the Pacific Ocean from Midway Island to China (east to west), and Australia to the U.S.S.R. (south to north). This is the strategic map on which you move your sub to find Japanese shipping lanes. Once you've found a convoy, the tactical map kicks in.

The tactical map is actually a series of three differently scaled maps. The Patrol Area Map is described above. You can zoom to the Navigation Map, which shows an area of 2,400 square miles around your sub, and zooming further yields the Attack Plot, a 40-square-mile area. The detail of land masses and enemy ships changes with the zoom. The Attack Plot displays the wake of each ship, to show which direction it is going. You use the Patrol Area Map to find the enemy convoy, the Navigation Map to close in on it, and the Attack Plot to position your sub for attack.

Superb Graphics

Once you've located your prey, you shift to a view of the conning tower, the captain's station. Using the joystick (the game is joystick- or keyboard-controlled), you either use the periscope or move the captain to one of the other stations: instruments and gauges, maps and charts, damage reports, quartermaster's log, or the bridge. Like the map screens, each battle station screen is graphically superb and very detailed.

Here are 79 reasons to buy at Elek-Tek, not to mention the fastest delivery anywhere.

I·MEGA
BERNOULLI BOX

1. 10 meg ½ height Drive for IBM-PC/XT/AT & compatibles ....$ 1675
2. 20 meg ½ height Drive for IBM-PC/XT/AT & compatibles 2335
3. Non Bootable Interface Card 104
4. Bootable Interface Card ...234
5. 10 meg cartridges for above (3 pak special) 125

Save 30% to 43%
off Manufacturer Suggested Ret. prices on
America's most wanted Printers

EPSON®

8. LX 80 ....$ 215
7. RX 100+ 299
8. FX 85 340
9. FX 286 ....CALL
10. LQ 1500 parallel ....CALL
11. DX10 Daisy Wheel 10CPS ...230
12. DX20 Daisy Wheel 20CPS ...320
13. DX35 Daisy Wheel 35CPS .590

Huge Discount on
TOSHIBA
OKIDATA star

UNBELIEVABLE!!!
XEROX/DIABLO D-36
DaisyWheel 35CPS
Mfr. Sugg. Ret. $1495
Elek-Tek Price $450

P-38 Dot Matrix 400CPS
Mfr. Sugg. Ret $1995
Elek-Tek Price $600

PRODUCTS FOR IBM-PC

14. **Amdek 310A** Amber Monitor ....$ 150
15. **Generic Multi** Multifunction Board 64K...135
16. **Generic Multi 384K** Multifunction Board, 384K..175
17. **AST Six Pak +** Multifunction Board, 64K ..225
18. **AST Six Pak + (loaded)** Multifunction Board 384K .290
18. **Quadram Quadboard** Multifunction Board, 0K ...195
20. Multi Board 64K/384K ..210/267
21. **Orchid Tech.** PC Turbo 186 570
22. ECCELL 0K 399
23. ECCELL Daughterboard...135
24. ECCELL I/O Board.........99
25. **Hercules** Monochrome Card.......299
26. **Hercules Color** Color Graphic Card 155
27. **Novation 4905921** 1200B Int. No Software....150 with MITE Software.......165
28. **Novation 490605-1** 2400BPS inc. Mite Software .620
29. **Novation 490603** ½ Card Modem 2400 BPS No software 425
30. **Novation 490603-1** As above inc. MS-DOS Software ..490
31. **Hayes 1200** External modem..........380
32. **Hayes 1200B** Internal modem w/software .359
33. **Hayes 2400** External modem...........599
34. **Hayes 2400B** Internal modem w/software .525
35. **AT&T 4000** 300/1200 Ext. Modem.....335
36. **US Robotics Courier 2400** Ext. 2400B Smart Modem .460
37. **Toshiba HND 04DT** ½ ht. DSDD Disk Drive.....90

38. Dysan
Internal H.D. Subsystem
20 Megabyte.........$500

39. I BAR 8-15
8 Outlet Surge Protector . $50

40. XIDEX Precision
DS/DD Diskette 10pk/$7.00
5 boxes minimum

DISKETTES

	Dysan	maxell	3M	MEMOREX	
3½" SSDD	20.00	20.00	23.00	—	—
DSDD	27.00	29.00	29.00	—	—
5¼" SSDD	15.00	13.00	12.00	11.50	11.00
DSDD	19.00	15.00	15.00	14.00	12.00
SSDD96TPI	24.00	24.00	24.00	—	—
DSDD96TPI	33.00	29.00	29.00	—	—
5¼" DSDDHD (For IBM AT)	33.00	28.00	30.00	—	24.00
8" SSDD**	22.00	29.00	25.00	—	—
8" DSDD**	26.00	32.00	29.00	—	—

**Unformatted

Call for Quantity pricing for 10 boxes or more.

3M DATA CARTRIDGES

73. DC100A.......$ 13.50
74. DC300A.........17.50
75. DC300XL........ 20.50
76. DC300XL/P.......21.50
77. DC600A 23.00
78. DC1000..........14.50
79. DC2000..........19.50

Call for quantity pricing for 10 cartridges or more.

CALL TOLL FREE 800-621-1269 EXCEPT Illinois, Alaska   CANADIAN TOLL FREE 800-458-9133

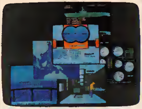
Corp. Accts. Invited. Min. Ord. $15.00. Visa or MasterCard by Mail or Phone. Mail Cashier's Check, Mon. Ord., Personal Check (2 wks. to clear) Add $4.00 1st item. (AK, HI, P.R., Canada add $10.00 first item) $1.00 ea. add'l shpg. & handl. Shipments to IL address add 7% tax. Prices subj. to change. WRITE for free catalog. RETURN POLICY: Defectives Only: Most products replaced within 30 days of purchase with identical merchandise only. Computer and large peripherals replaced only when defective on arrival (within 3 work days of delivery). Other problems covered by mfr. warranty. ALL ELEK-TEK MERCHANDISE IS BRAND NEW, FIRST QUALITY AND COMPLETE. Delivery subject to availability. DUNS #06-719-6617

ELEK-TEK, inc. 6557 N. Lincoln Ave. Chicago, IL 60645
(312) 631-7800   (312) 677-7660

If the sub is surfaced, you can climb to the bridge. From here you can look around to spot the enemy ships, using your naked eyes or binoculars. The view includes the Japanese ships (several varieties, each graphically different) and the land on the horizon. The landforms correspond exactly to where you are in the Pacific; if you patrol to the coast of New Guinea, you will see the coast of New Guinea. Given the amount of territory covered in the game, the mapping system is obviously very sophisticated.

The binoculars and periscope screens include all the details necessary for firing at the ships: target type, target range and speed, and such data as angle-on-bow. You may fire the deck guns or torpedoes, but only torpedoes are consistently effective.

These screens are the heart of the action, but to keep the sub running you must pay strict attention to your vessel's instruments and gauges. They display information about the battery, the depth of the sub and of the ocean floor beneath you, fuel levels, the status of hull openings, and so on. For instance, the battery allows restricted underwater maneuvering, depending on your speed, and then must be recharged on the surface.



This assortment of screens from Silent Service *shows the diversity of the program's graphics.*

Attention To Detail

Other facets of the game are equally realistic. If your sub hits the ocean bottom, you hear a scraping sound and the hull may be damaged. You can cruise at four speeds or cut the engines for silent running. More esoterically, once per mission you can get rid of your emergency tanks to stop a fatal dive, or release debris to the surface to fool the Japanese destroyers into thinking you've been sunk. Sound effects range from sonar pings—telling you that destroyers are closing in—to the ominous explosions of nearby depth charges. The sub's hull even creaks if you dive deeper than it was tested for, and you hear a grinding metallic sound if you're rammed by an enemy ship. *Silent Service*'s detail is astonishing.

But the most impressive part of the detail is that it does not impede play. Detail in the more traditional type of war game frequently hinders understanding and lengthens the game considerably, but *Silent Service* plays quite easily after only a half-hour or so of practice. Once you learn how to steer the sub and fire torpedoes at a target, you can try a mission. You can learn the rest, such as diving and running silent at the approach of a Japanese destroyer, as the situation demands. At any time, you can pause the game to allow you time to think.

Like *F-15 Strike Eagle, Silent Service* is both intriguing and addicting. Also like *F-15*, it is highly educational, but there is nothing tedious about the lessons. The excellent manual describes the submarine war in the Pacific, the background to the scenarios, and the tactics used by sub captains. By playing the game, you'll quickly find that these captains knew what they were doing. *Silent Service* is a superior product.

Silent Service
MicroProse Software Inc.
120 Lakefront Drive
Hunt Valley, MD 21030
$34.95–$39.95 (depending on version)

DeluxePaint For Amiga

Lee Noel, Assistant Editor, Art & Design

Requirements: Amiga with at least 256K RAM (512K recommended). Printer optional.

Whenever a new computer appears on the market, some of the most important factors affecting its success are the quality and diversity of its software. In the case of the long-awaited and innovative Amiga, questions concerning software support become even more important. Is this computer the powerhouse it's said to be, and can programs be written to take full advantage of its capabilities?

Electronic Arts, a software publisher widely considered to be at the forefront of personal computing, said yes to both questions and threw its considerable weight squarely behind the Amiga. *DeluxePaint,* by Dan Silva, is one of the first results. Not surprisingly—considering the Amiga's selling point as a computer for those who want a "creative edge"—*DeluxePaint* is a visual arts program of immense scope and flexibility.

In fact, *DeluxePaint* is really three different programs of immense scope and flexibility. Due to differing memory requirements, *DeluxePaint* includes a separate program for each of the Amiga's three major screen modes: 320 pixels across by 200 down with 32 simultaneous colors; 640 × 200 with 16 colors; and 640 × 400 with 16 colors. The number of simultaneous colors in each mode can be selected from a palette of 4,096 possible colors. You can also customize *DeluxePaint* by restricting it to a smaller palette.

After booting up the program disk, you must type in a command to call up whatever incarnation of *DeluxePaint* you want. This may sound confusing to nontechnical artists hoping to use the Amiga for their first experiments in computer graphics, but loading the program is fairly straightforward. First you turn on the Amiga and insert the usual Kickstart disk. When the prompt asking for the Workbench disk appears, you insert the *DeluxePaint* disk instead. AmigaDOS comes up next with its 1> prompt, and then you type the appropriate command. For instance, you'd enter dpaint and press RETURN to work in the 320 × 200 mode.

Best For 320 × 200

DeluxePaint works best by far in the 320 × 200 mode. In the 640 × 200 mode, pixels are three times as high as they are wide, and the program slows down considerably. In 640 × 400, the slowdown is drastic. So much memory is consumed that there's not enough room in a 512K machine for both the entire program and a screen. Instead, the program is broken into modules that are constantly swapped in and out from disk. (*DeluxePaint* works this way in all modes on a 256K Amiga.)

Also, the 640 × 400 mode suffers from a jittering screen display. The jittering varies depending on the color combinations, and high-contrast combinations are worse. This isn't *DeluxePaint*'s fault—the monitor simply cannot refresh the 256,000 pixels in this mode fast enough to display a stable picture. (Other computers with similar modes get around this problem by using special monochrome monitors driven at higher refresh rates.)

Since *DeluxePaint*'s features are the same in all modes, we'll describe what's available in the 320 × 200 mode. This is the most color-rich screen, and the program's documentation and all of the sample pictures on the disk are slanted toward it.

kyan

ALL NEW SOFTWARE LINEUP!

kyan pascal (Version 2.0) $69.95

kyan pascal is the ideal system for learning Pascal and developing Pascal programs. It's a full implementation of ISO Pascal and conforms to the standards set by the Federal Software Testing Center. ***kyan pascal*** features a menu-driven environment with multiple HELP screens; a full-screen text editor; and, optimized 6502 machine code compiler/assembler. It produces code that runs at the maximum speed possible on the 6502 microprocessor. ***kyan pascal*** supports many extensions including string hendling, linking, chaining, random files, and included or inline assembly source code. It also supports a line of powerful toolkits which make it possible for even novice programmers to develop sophisticated software. ***kyan pascal*** (Version 2.0) requires only one disk drive. It is available for the Apple II (runs in ProDOS and requires 64K); Atari (runs DOS 2.5 and requires 48K); and Commodore 64/128. ***kyan pascal*** is not copy protected and comes with a Pascal tutorial end reference guide.

Programming Utility Toolkit $49.95
Progremming is faster and easier with this extensive library of utility programs and file management procedures. The Toolkit includes source code for more than 20 utility programs.

Advanced Graphics Toolkit $49.95
Add stunning graphics to your Pascal or assembly language programs. With the Toolkit's graphics primitives, you can build a custom graphics library. Or, you can use the Toolkit's library for 2 and 3 dimensional transformations, windows and clipping, shading, end more.

MouseText Toolkit** (available for Apple II only).. **$49.95
Add Macintosh-like graphics to your Pascal programs. The Toolkit includes routines for pull-down menus, windows, and mouse-controlled cursor events (Toolkit requires Apple IIc or enhanced IIe).

Macro Assembler/Linker $69.95
kyan's latest programming tool adds a new dimension to assembly language programming. The Assembler/Linker includes a text editor, 65C02 macro assembler, object module linker, debugger, and librarian.

kyan Software offers you a 15 day money back guarantee. See for yourself . . . kyan is the best programming software.

Send Check/Money Order: kyan software, Dept. P • 1850 Union Street, #183 • San Francisco, CA 94123

Or Call: (415) 626-2080 • Visa/MC Accepted

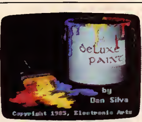
Please include $4.50/order for shipping/ handling, $12 outside North America CA residents add 6.5% sales tax

Tons Of Tools

Like virtually all Amiga software, *DeluxePaint* is a mouse-driven, icon-based program, similar in some ways to *MacPaint* for the Macintosh. An array of drawing tools is represented by icons on the computer's display (the tools can be hidden when the picture is finished). Tools are selected by moving a pointer to the appropriate icon with the mouse. A click of the mouse button activates the tool, which can then be used in the drawing area.

DeluxePaint has practically all of the tools that have become standard in graphics-design programs. You can draw straight lines and a multiplicity of outlined or filled shapes, paint with different-sized brushes or an "airbrush," print text on the screen, and lots more. But *DeluxePaint* really shines because it offers unique new tools and novel extensions to the old standbys.

First and foremost, *DeluxePaint* recognizes the importance of color to the artist. The program has a special palette window where any of the Amiga's thousands of colors can be mixed and used in design work. The palette appears from the program's title bar as a pull-down menu (the menu selections have alternate keyboard commands as well.)



This picture of a paint can and brush—included on the DeluxePaint *disk—shows the fine shading possible with 32 simultaneous colors chosen from a palette of 4,096.*

Within the palette is an almost bewildering number of options. For instance, there are two ways to make color adjustments. The first method has three slide controls—manipulated with the mouse—that alter the percentages of red, green, and blue in any color (these are the primary colors for a video display). Another set of three sliders allows changes to the hue, saturation, and value of any selected color. The latter system is much like the tint, color, and contrast controls on a normal color TV. As a result, novice users of *DeluxePaint* may find this system reassuringly familiar.

If that's not enough flexibility, the artist can also move the palette window to any convenient location, and the relocation will be "remembered" for the rest of the current session.

Flowing Colors

Colors are selected simply by pointing and clicking with the mouse; selection is verified by a highlighted box. Color changes are instantly reflected in the palette window and in the picture. As a consequence, it's delightfully easy to adjust colors relative to each other. The program disk includes two good examples of the effects made possible by this precise control over a diverse palette. "KingTut" displays the hard, gleaming gold coffin mask of the Egyptian boyking Tutankhamen. In contrast, "Venus" faithfully reproduces the soft, almost pearly hues of Botticelli's "Birth of Venus."

Some really amazing special effects are also built into *DeluxePaint*'s palette, such as animated color cycling. This allows the artist to establish three sets of colors that will cycle through a certain range. Each range can be narrow or wide, can include harmonious or clashing colors, and can overlap the

COLOR MONITOR

Sale

SALE!!!

16 COLORS

(Premium Quality)

- **Built In Speaker & Audio**
- **For Video Recorders**
- **For Small Business Computers**
- **Apple - Commodore -Atori - Aplus 3000 -etc.**
- **One Yeor Free Immediate Replocement Worronty***

(Premium Quality)

- **Beautiful Color Controst**
- **High Resolution**
- **Shorp Clear Text**
- **Anti-Glore Screen**
- **40 Columns x 24 Lines**
- **Front Ponel Controls**

List $329⁰⁰

Sale $159⁹⁵*

Add $14.50 Shipping

80 Columns

RGB

Super High Resolution

13" Color Computer Monitor*

*C64/Atori composite coble $9.95

* C128 RGB/Composite 80 column coble $19.95.

13" RGB & COMPOSITE COLOR MONITOR

Allows use of C-128 ond C64 mode - composite ond 80 column RG8 mode. Must be used to get 80 columns in color with 80 column computers. Specioly designed for use with the C128's speciol composite video output, plus green screen only option switch. (odd $14.50 shipping)

List $399.00

Sale $259⁹⁵*

12" MAGNAVOX (NAP) 80 COLUMN MONITOR

Super high resolution composite green screen monitor. 80 columns x 24 lines, eosy to reod, plus speoker for oudio sound included. Fontostic volue. Limited Quontities.

List $129.00 **Sale $79⁹⁵***

Turn Your Monitor into a TV Set Without Moving Your Computer

Elegant TV Tuner with duol UHF/VHF selector switches goes between your computer ond monitor. Includes mute, outomotic fine tuning ond computer-TV selector switches. Inputs included for 300 ohm, 75 ohm, ond UHF. Con be used with coble TV ond VCR's. Fontostic Volue. Limited Quontities.

List $129.95 **Sale $49⁹⁵***

15 Day Free Trial - 90 Day Immediate Replacement Warranty

• LOWEST PRICES • BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 far shipping hondling ond insuronce Illinais residents please odd 6¼% tox Add $20 00 for CANADA PUERTO RICO, HAWAII, ALASKA APO-FPO orders. Conadian orders must be in U.S dollors WE DO NOT EXPORT TO OTHER COUNTRIES EXCEPT CANADA Enclose Coshier Check, Money Order or Personal Check Allaw 14 doys far delivery, 2 ta 7 days for phone orders, 1 doy express moil! Prices & Avoilobility subject to change without notice
VISA — MASTER CARD — C O D   No C O.D. ta Canada APO-FPO

COMPUTER DIRECT
We Love Our Customers
22292 N. Pepper Rd., Borrington, Ill. 60010
312/382-5050 to order

240K Apple® Compatible Computer System

NEW SALE

APlus 3000 computer system includes 192K RAM, 48K ROM (32K Micrasaft Basic plus 16K ROM Emulotor), 160K Laser 5¼" Disk Drive (Runs Apple II Software), Magic Window Wardprocessor, MagicCalc spreadsheet, Magic Memory Database. All for only $399.00

Complete System

$399⁰⁰

• 15 Day Free Trial

Double Immediate Replacement Warranty

If any of the Aplus 3000 computer system equipment fails due ta faulty workmanship ar material within 180 days of purchase we will REPLACE it immediately with na service charge!!

• Over 10,000 existing Apple® programs • Centronics printer interface included
• 240K (192K RAM, 48K ROM) • ArtSci's Magic Windaw II, Magic Memary, and MagiColc included
• 160K Loser 5¼" Disk Drive (Runs Apple II software) • RGB (80 columns in color) and compasite included

SPECIFICATIONS

A plus 3000 is o complete, self-contained computer bosed on the popular 6502A microprocessor and can tap into the tremendous software library of Apple II. Feotures include 192K 8ytes RAM, 32KB Enhanced Microsoft BASIC, 80 column text, 560H X 192V color graphic display, 81 key sculptured keyboard and high efficiency switching power supply. Also included os standard are Centronics bus printer interface, Cassette interface, 4 channel sound generator, ond 5¼" Apple Compatible Disk Drive.

• TEXT

— 40 columns X 24 rows or 80 columns X 24 rows software selectable.
— 5 X 7 choracters in 7 X 8 matrix.
— Upper and lower case characters.
— One of Eight colors for characters/graphics and background, Red, Green, Blue, Cyan, Magenta, Yellow, 8lack and White.
— Character set with normal, inverse and flashing capabilities.

• GRAPHICS

— 280H X 192V 6 colors — Black, White, Violet, Green, Blue, Orange.
— 280H X 192V 8 colors bit image — Black, White, Red, Green, Blue, Cyan, Magenta, Yellow.
— 560H X 192V 6 colors — Black, White, Violet, Green, Blue, Orange. (High resalution color monitor required)

More Features than Apple® for less than Commodore®

Features	**Aplus 3000**	Apple IIe	Commodare C-128
RAM	192K	64K	128K
Runs Apple II Software	Yes	Yes	No
Function Keys	24	None	16
4 Voice, 6 Octave Sound	Yes	No	Yes
Composite Video	Yes	Yes	Yes
Disk Drive	included	Extra Cost	Extra Cost
Numeric Keypad	included	Extra Cost	Included
Video Cable	included	Extra Cost	Extra Cost
RGB Color Card	included	Extra Cost	Included
80 Column Card	included	Extra Cost	Included
Centronics Printer Interface	Included	Extra Cost	Extra Cost
Drive Controller	Included	Extra Cost	Included
$150 Wordprocessor (Magic Window)	Included	Extra Cost	Extra Cost
$150 Spreadsheet (MagiCalc)	Included	Extra Cost	Extra Cost
$60 Database prg. (Magic Memory)	included	Extra Cost	Extra Cost
Your Cost	**$399.00**	$1745.00	$1117.90

Super Apple Compatible Disk Drive Sale $149.95.
Quieter, Cooler, Better Disk Drives for your Apple II plus, IIe, IIc (specify when ordering). List $299.95. Sale $148.85.

15 Day Free Trial — If it daesn't meet yaur expectatians within 15 days af receipt, just send it back ta us UPS prepaid and we will refund yaur purchase price!!

ACCESSORIES	LIST	SALE
2nd Disk Drive	$299.95	$149.95
2 professional analog joysticks	$ 39.95	$ 24.95
Z-80 cart. ollaws CP/M use	$ 99.95	$ 59.95
RS232 adapter	$ 99.95	$ 59.95
R/F Modulotor (TV haakup)	$ 29.95	$ 19.95
RGB cable (RGB Monitar haokup)	$ 24.95	$ 19.95
Centronics cable (for Centronics printer)	$ 34.95	$ 24.95
Technicol reference manual	$ 29.95	$ 19.95
80 columns Hi-Res Green Monitor	$199.aa	$ 79.95
80 column Hi-Res RGB Manitar	$399.00	$259.00

Add $25.00 far shipping and handling!!

Enclose Coshiers Check, Money Order or Personol Check Allow 14 days for delivery 2 to 7 days for phone arders. 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only Add $10 more if C.O.D.

COMPUTER DIRECT

We Love Our Customers
22292 N. Pepper Rd., Barringtan, Ill. 60010
312/382-5050 to order

APPLE and COMMODORE are registered trademarks of Apple Computer Inc. and Commodore Business Machines Inc. respectively

Famous Smith Corona National Brand

10" PRINTER SALE

Below Wholesale Cost Prices!!!

• ONE YEAR IMMEDIATE REPLACEMENT WARRANTY

- **Speed: 120 or 160 characters per second**
- **Friction Feed/Tractor Feed — Standard**
- **80 character print line at 10 CPI**
- **1 Line Buffer, 2K Buffer on 160 CPS Plus LQM**
- **Six pitches**
- **Graphics capability**
- **Centronics compatible parallel interface**
- **Features Bidirectional Print, Shortline Seek, Vertical And Horizontal Tabs**

SUPER GRAPHICS

```
This is a sample of our           emphasized
near-letter-quality print.

italic print.   There is standard data
                processing quality print
```

Check These Features & Prices

120 CPS 10" Printer

List $429.00

SALE $159

160 CPS + Letter Quality Mode 10" Printer

List $499.00

SALE $199

SPECIFICATIONS

(IBM — Commodore)

Size/Weight
Height 5.04" Width 16.7"
Depth 13.4" Weight 18.7 lbs.
Internal Char. Coding
ASCII Plus ISO
Print Buffer Size
120 CPS: 132 Bytes (1 line)
120/160 CPS Plus LQM: 2K
No. of Char. in Char. Set
96 ASCII Plus International
Graphics Capability
Standard 60, 72, 120 DPI
Horizontal 72 DPI Vertical
Pitch
10, 12, 16.7, S, 6, 8.3, Proportianal Spacing
Printing Method
Impact Dot Matrix

Char. Matrix Size
9H x 9V (Standard) to 10H x 9V
(Emphasized & Elangate)
Printing Features
Bi-directional, Shart line seeking, Vertical Tabs, Harizontal Tabs
Forms Type
Fanfold, Cut Sheet, Roll (aptianal)
Max Paper Width
11"
Feeding Method
Friction Feed Std., Tractor Feed Std.
Ribbon
Cassette — Fabric inked ribbon
Ribbon Life
4 millian characters

(Apple — Atari — Etc.)

Interfaces
Parallel 8 bit Centronics compatible
120/160 CPS Plus NLQ: RS232 Serial inc.
Character Mode
10 x 8 Emphasized; 9 x 8 Standard; 10 x 8 Elongated; 9 x 8 Super/Sub Script (1 pass)
Character Set
96 ASCII
11 x 7 Internatianal Char.
Line Spacing
6/8/12/72/144 LPI
Character Spacing
10 cpi narmal; S cpi elongated narmal; 12 cpi compressed; 6 cpi elongated compressed; 16.7 cpi condensed; 8.3 cpi elangated candensed, 5.12.S cpi elongated propartianal

Cartridge Ribbon — List $19.95. **Sale $12.95**.

Interfaces

IBM $89.00	**Apple $59.00**	**Atari $59.00**	**Commodore $39.95**

Add $14.50 for shipping handling and insurance Illinois residents please add 6¼% tax. Add $29.00 for CANADA, PUERTO RICO, HAWAII ALASKA APO-FPO orders Canadian orders must be in U S dollars. WE DO NOT EXPORT TO OTHER COUNTRIES, EXCEPT CANADA Enclose Cashiers Check, Money Order or Personal Check Allow 14 days delivery 2 to 7 days for phone orders 1 day express mail! Prices & Availability subject to change without notice
VISA — MASTERCARD — C O D No C O D to Canada or APO-FPO

COMPUTER DIRECT
We Love Our Customers
22292 N. Pepper Rd., Barrington, Ill. 60010
312/382-5050 to order

ATARI®

152K *Lowest Price In The USA!* 152K

Computer System Sale

• Students • Word Processing • Home • Business

LOOK AT ALL YOU GET FOR ONLY **$379** **SYSTEM PRICE**

LIMITED QUANTITIES

	LIST PRICE	INDIVIDUAL SALE PRICE
① Atari 130XE 152K Computer	$249.00	$\$134^{95}$
② Atari 1050 127K Disk Drive	299.00	159^{95}
③ Atari 1027 Letter Quality 20 CPS Printer	299.00	159^{95}
Atari Writer Plus Word Processer with Spell Checker	59.95	49^{95}
Atari BASIC Tutorial Manual	16.95	12^{95}
TOTALS	**$923.90**	**$517.75**

All connecting cables & T.V. interface included.
☆ Monitors sold separetly.

SAVE OVER $100
All 5 ONLY
$\$379^{00}$
SYSTEM SALE PRICE

CALL FOR 1027 PRINTER REPLACEMENT OPTIONS

Other Accessories	List	Sale	
☆ 12" Hi Resolution Green Screen Monitor	$199.00	$79.95	Add $9.95 for Connection Cables
☆ 13" Hi Resolution Color Monitor	$399.00	$159.95	Add $10 for UPS

15 DAY FREE TRIAL. We give you 15 days to try out this ATARI COMPUTER SYSTEM!! If it doesn't meet your expectations, just send it back to us prepaid and we will refund your purchase price!! **90 DAY IMMEDIATE REPLACEMENT WARRANTY**. If any of the ATARI COMPUTER SYSTEM equipment or programs fail due to faulty workmanship or material within 90 days of purchase we will replace it IMMEDIATELY with no service charge!!

Best Prices • Over 1000 Programs and 500 Accessories Available • Best Service • One Day Express Mail • Programming Knowledge • Technical Support

Add $25.00 for shipping and handling!!
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only. Add $10 more if C.O.D., add $25 if Air Mail.

COMPUTER DIRECT
We Love Our Customers
22292 N. Pepper Rd., Barrington, Ill. 60010
312/382-5050 to order

ranges for the other two cycles. Once activated, a cycle runs through all the colors in its range in a smooth sequence. The speed of each cycle can be individually controlled with the mouse and a slider. Color cycling is what makes the disk's sample waterfall picture seem to flow. With thought and care, you can create effects otherwise impossible in two-dimensional art.

Closer to traditional art media are tools which smear, shade, and blend. Smearing enables an artist to use the *DeluxePaint* brush to smudge colors already painted on the display. The effect is similar to running a brush through fresh oil paints. Shading and blending work on a defined color range, like the ranges for color cycling. Both operate best on a range of closely related shades and affect only the colors in the selected range.

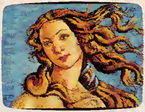
The action of shading is difficult to explain in print, but blending does pretty much what it describes—it produces smooth gradations like those found in watercolor washes or airbrushing. There are also tools for exchanging colors and an undo feature to recover from mistakes.

Anything Is A Brush

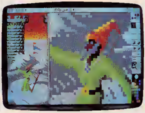
All that's fine for the colors, but what of the brushes that apply them? Once again, *DeluxePaint* offers abundant options. There are, of course, some built-in brushes. These are various useful shapes, and they can all be adjusted in size with one of the program's easy-to-use tools. But, as might be expected, other brush manipulations range into some wild and unexplored regions.

The essence of *DeluxePaint* brushes is this: Anything can be a brush, so you can paint with a single pixel, a pictorial element, a whole screen, or anything in-between. And if that's not enough, the artist can grab the brush, resize it, rotate it to any angle, flip it, or drag it into a completely new shape. This last feature allows a brush that looks flat to be made to appear three-dimensional. To get an idea of the effect, imagine a flag lying flat on a table. Suddenly, it's bent into a stiff billow and paints in an arc across the sky—stars and stripes and colors and all.

Brushes, like complete pictures, can be saved to previously prepared data disks. There are also some esoteric possibilities involving the exchange of transparent and solid colors within a brush, plus a different way to "hold" the current brush, and a way to speed up response time to certain types of brushes. Analogies are hard to come by for these features, but they open up fascinating possibilities not available to artists working in traditional media.



Another sample picture on the DeluxePaint *disk is this reproduction of Botticelli's "Birth of Venus."*



This photo demonstrates a zoom window, just one of the many features in DeluxePaint.

The basis for all the brush transformations is the designer's ability to reach out and capture any area of the screen with a special brush selection tool. This is much like the copy, cut, and paste functions found in other graphics programs, and it can be used for those purposes as well as brush design.

Room To Zoom

A full description of *DeluxePaint*'s myriad features would run on for many pages (the manual is 31 pages long), but highlights of some of the major ones not covered so far bear mentioning.

DeluxePaint has a special magnification tool that allows the artist to zoom closer and closer to the area under inspection, and then back away in similar increments. Great mobility within the magnify mode is provided with the cursor keys.

There's fairly complete printer support, plus the ability to add text to designs and manipulate it in numerous ways. A skewing feature even lets you turn ordinary text into italics.

For precision design work, one option gives a constantly updated display of the cursor's screen coordinates, another provides a grid that can be modified, and still another allows unusual mirror-like symmetry effects.

Virtually anything that anyone ever wanted in a personal computer graphics program is included in *DeluxePaint*—and it's all easy to use and easy to learn. It's fortunate that the program is fairly intuitive because the documentation is not. The basic tone of the manual is that experimentation and playfulness are the best methods for coming to grips with the program. A step-by-step approach might have been more helpful. And, ironically, the manual's graphics are almost nonexistent.

Another problem with the manual is that it lacks completeness. *DeluxePaint* cannot create data disks by itself, so the artist must refer to Chapter 4 of the *Amiga User's Guide* for the information. Also, hardly any program commands are summarized in *DeluxePaint*'s command summary.

On the other hand, in the few places where you might get really stuck, the documentation comes through with some solid tutorials.

Bottom line: Will *DeluxePaint* and an Amiga give you that creative edge? That depends—in the end, it's still the artist that has to pull the rabbit out of the hat.

DeluxePaint
Electronic Arts
2755 Campus Drive
San Mateo, CA 94403
$79.95 (Introductory price)
$99.95 (Regular price)

S'More For Commodore 64

Art Hunkins

Requirements: Commodore 64 or a Commodore 128 in 64 mode.

Commodore 64 owners who wish to upgrade their computers have two main options: Buy a Commodore 128 or install a *S'more* cartridge from Cardco. Each choice has its advantages. If money is no object (and you aren't overly attached to your 64), you might consider the 128. But the choice isn't that clear-cut. For those who write their own BASIC programs, *S'more* has some significant advantages of its own. Frankly, it's difficult to know which to compare *S'more* to—the 64 without *S'more*, or the 128.

Of course, the 128 does have some things going for it: twice the available user memory (122,365 bytes) as the 64; BASIC 7.0, with powerful commands for graphics, sprites, sound, and windowing; and a FAST mode for double-speed operation. So if it's raw computer power and extra memory you want, the 128 is hard to beat.

On the other hand, *S'more* BASIC is more comprehensive than BASIC 7.0 in its utilities; it defaults to disk LOAD, offers a greater variety of input options as well as more flexible screen formatting, and includes varied reset options. The built-in utilities are a real boon: MERGE, AUTO, HEX, DEC, FIND, CHANGE, reNUMBER, DUMP, and OLD—all familiar to BASIC AID users. The LIST command can scroll up and down, not true of BASIC 7.0. On the 128, only AUTO, RENUMBER, and a disk file APPEND are implemented.

Compared to the unenhanced 64, *S'more* frees up 57 percent more user memory—61,183 bytes instead of 38,911 bytes. The memory is contiguous and can be used in any way you desire. (As we'll see, there are other protected locations where machine language routines up to 512 bytes long may be stored.)

Improved Disk Commands

S'more BASIC and BASIC 7.0 come out about even when it comes to disk commands (a notable weakness with the unexpanded 64); only the approach is different. Whereas 7.0 gives a wealth of specific commands, *S'more* uses only one—DISK, an all-purpose "wedge" followed by the traditional disk access symbols. Both BASICs also offer numerous enhancements of standard commands (such as a RUN that LOADs and RUNs a BASIC program from disk). Both permit the SHIFT-RUN key combination to LOAD/RUN the first program on disk.

Both BASICs offer about the same range of programming structures (DO-LOOP, WHILE-UNTIL, IF-THEN-ELSE). Both implement error-trapping and HELP, and both have programmable function keys, though 7.0 sets aside almost twice the buffer (246 bytes versus 128) for key definitions.

S'more is also handy in that its LOAD and SAVE commands default to disk (there is no DLOAD or DSAVE), and that it includes a disk CATALOG/directory option. In fact, due to the way the disk default option works, you can display the CATALOG, cursor to the program you want, type LOAD (or RUN), and hit RETURN—without worrying about what is displayed after the program name.

ML Limitations

For BASIC programs, *S'more* is superb. But let's look at ML applications. Here the picture is not so clear.

Although *S'more* has a MONITOR command, it doesn't have a built-in monitor; MONITOR just links you to a monitor if you've loaded one into memory. *S'more* comes with a disk of software that includes a version of *Micromon* called *Smon*. (Other programs on the disk illustrate applications of the more noteworthy *S'more* BASIC extensions.)

Cardco's manual is thorough, clear, instructive, and particularly forthright when it describes *S'more's* limitations with memory addressing and machine language. Here's the catch: To make so much contiguous BASIC memory available, Cardco had to change a lot of memory locations and reconfigure memory. Cardco did what it could to maintain compatibility with Commodore 64 BASIC (BASIC 2.0), but there were limits on what was possible.

It's remarkable that low memory with *S'more* is so highly compatible with BASIC 2.0. Only two differences will be noticed by the average programmer. First, and most importantly, the cassette buffer has been moved. ML programs designed to reside there will have to be transported to the new location. Also, some of the previously free bytes (which you may have used for flags or temporary data storage) are free no longer (zero page 251–254 remain available, however). There is a bonus, though—a 512-byte RS-232 input/output buffer, protected from BASIC, which can be utilized for ML routines in most cases.

The most critical low memory locations for the BASIC programmer, the keyboard buffer and its corresponding character counter, remain intact. As the manual clearly states, however, ML routines that access ROM are in for major rewrites. The only ROM routines that are safe to use are the Kernal routines when they are accessed through the vectors in low memory (these vectors are unchanged in location). You cannot access ROM subroutines directly. This is a problem particularly with the SID, VIC, and CIA chips—that is, when working directly with screen, sound, and input/output peripherals.

The *S'more* Solution

To get around these limitations, the manual suggests that perhaps most ML routines are best written in *S'more* BASIC, then compiled with the (not-yet-released) *S'more* BASIC Compiler. This suggestion indicates the degree of potential difficulty in converting most ML programs for use with *S'more*.

But there's another alternative, too. *S'more* establishes a set of CIA, VIC, and SID *reserved variables* (DIMensioned arrays). Each variable corresponds to a CIA, VIC, or SID chip location you might wish to PEEK or POKE. To POKE the location, just assign the variable the desired value; presto, the POKE is done. To PEEK the location, just use the reserved variable in an expression. It works fine and is simpler than actually PEEKing and POKEing. For sound and the SID chip, for example, it is not too far from the convenience of using BASIC 7.0's new sound commands (PLAY, FILTER, ENVELOPE, etc.)

Of course, this technique works only from BASIC, not machine language. There are times when, for speed and efficiency, ML is required. Although conversion of ML routines accessing the support chips is possible, it is apparently far from trivial. (The manual does not attempt to explain; it only hints that RAM/ROM bank-switching is involved, and that the banking system is similar to that of the Commodore Plus/4.)

There is but one other limitation I've noticed with *S'more.* When writing or editing a BASIC program, the enhanced BASIC often responds slowly, particularly with long programs. The cursor can take 1.5 to 2.5 seconds to reappear after you hit RETURN to enter a new line; it takes longer toward the beginning than at the end of a program. On the other hand, garbage collection purportedly is speeded up dramatically over 2.0 BASIC.

Put up your dukes!

And your Earls and Viscounts. If you've got royal ancestors, we have the noble software that can help you trace them down.

Family Roots and your Apple, IBM, Commodore, Kaypro*, and many others, offer individual and group sheets, charts, name indices, general search and text capabilities. Adapts to most disk drives, printers, and screens. You get more utility programs, plus lots of personal control. A comprehensive (new) manual is included.

All for just $185.

Write or call today for more information and a free brochure.

Family Roots

Quinsept, Inc.

P.O. Box 216
Lexington, MA 02173
(617) 641-2930

American Express, Visa, and MasterCard gladly accepted.

* *Trademarks for Apple Computer Inc, International Business Machines, CBM, Inc and Digital Research*

Works With 128, Too

These are the only problems I've experienced working with the *S'more* cartridge. Overall, *S'more* maintains a high degree of compatibility with BASIC 2.0 (and its associated memory configuration), offers more than 50 percent additional memory accessible to BASIC, and a greatly enhanced language. It makes working with the screen and sound a simpler task for BASIC programmers.

In short, *S'more* is a cost-effective alternative to a Commodore 128 upgrade. (Cardco's literature describing *S'more* as a "bridge to the 128" is on target.) And even if you do decide later to acquire a 128, *S'more* works identically on the 128 in 64 mode.

S'more
Cardco, Inc.
300 S. Topeka
Wichita, KS 67202
$69.95

Heart Of Africa

Neil Randall

Requirements: Commodore 64 or 128 in 64 mode; Apple II-series computer with at least 64K RAM; or an Atari 400/800/XL/XE with at least 48K RAM. Disk only.

One of the truths in the entertainment industry is that anything popular will spawn many imitators. The field of computer games is certainly no exception. The first hits were *Space Invaders*, then *Pac-Man*, and then *Donkey Kong*. Each of these games begat a host of imitations, few of which approached the quality of the original. Imitations are rarely as good as the things they imitate.

Still, for every imitator trying to capitalize on the popularity of someone else's game, there's a designer trying to improve upon his own original design. This is not imitation—at least not what we normally think of as imitation. Instead, the designer is making an honest effort to improve on a game concept. Much like an artist, who chooses a medium and produces work after work trying to perfect his use of that medium, the game designer invents a system, then produces game after game to develop the system to its fullest. As long as the system keeps improving, the enterprise is justified.

Such is the case with Ozark Softscape's *Heart of Africa*, published by Electronic Arts. *Heart of Africa* is an extension of the system pioneered in Ozark's own *Seven Cities of Gold* (reviewed in the September 1984 issue of COMPUTE!). Far from an imitation, it improves on the original game and offers a fresh approach to a system that many people considered near-perfect already. Both games deal with exploration, but *Heart of Africa* gives us something more: a quest.

In Search Of A Tomb

Your quest in *Heart of Africa* is to find the lost tomb of Ankh Ankh, somewhere in the middle of the Dark Continent. You travel alone, buying supplies and tools wherever you can. As you cross the continent, you make discoveries and try to obtain clues about the lost tomb from tribal chiefs. It's not hard to get information, but it's very hard to get useful information, and just as hard to stay alive. The perils are constant, from dying of thirst in the Sahara Desert to suffering a fatal bite by a poisonous snake.

Like *Seven Cities of Gold, Heart of Africa* is entirely joystick-driven. You can put your feet up, lean back in your easy chair, and play the game without touching the keyboard. For further playability, the game offers a diary that continually updates itself. The diary is a graphically attractive series of pages that records special events. On the surface, it seems only a nice addition to the game, but in play it greatly eases record-keeping. Any exploration game, be it a text or graphics adventure, demands some keeping of records: mapmaking, recording conversations, jotting down clues. But *Heart of Africa* takes most of these out of your hands. The map is produced for you on the screen, and your observations, even conversations, are recorded in the diary. You can read the diary at any point simply by loading it from disk. It makes the game extremely playable, especially for those who loathe keeping records.

The *Heart of Africa* game screen shows a solitary figure marching across the map. As you walk, the map scrolls north, south, east, or west, shedding light on more and more of the Dark Continent. The map is constantly updated, and you can check it at any point during the game to see what you've already discovered. As you travel, you discover villages, mountain ranges, rivers, lakes, and, of course, if you work hard enough, the source of the Nile.

Random events are sometimes positive, such as finding valuable caches left behind by previous explorers, as well as negative, such as encounters with crocodiles, poisonous snakes, or rhinoceri. If you're equipped with the right weapons, you can normally stave off an attack, but you may become ill, fatigued, or very thirsty. You can paddle a canoe along the rivers and lakes, and you can even go over waterfalls. The entire continent is yours to discover.

Tribal Relations

Perhaps the most impressive part of the game is the interaction with the tribes. As in *Seven Cities of Gold*, where cooperating with the natives established your reputation, working with the tribes in *Heart of Africa* is difficult. Each tribe is different and each chief reacts differently to you. For some tribes, a few gifts will yield helpful information. For others, all the gold in the world seems insufficient. You can steal supplies by wielding your gun, but your reputation will suffer. Or worse, you may catch a blow dart. The only way you can know how a tribe will react is to visit each village. If you do well and reward the chief, he'll tell you what else you might bring for more information. If you do poorly, you'll be drummed out of the village.

The *Heart of Africa* manual consists primarily of the notes written by your predecessor, the person sending you on this mission. It describes each of the areas of Africa and the tribes therein. An impressive document for its sheer information, it is also vital for gaining clues about where you should go. It gives, for instance, translations of the tribal names for geographical points. To the natives, after all, Victoria Falls is not Victoria Falls.

There is nothing easy about the game, but the difficulty comes from the situation, not in trying to learn the system. It is extremely easy to get across Africa, buying things, finding things, and giving things away, but it is very hard to gain useful information. Still, this is the game's strength. A poor game is difficult to learn and offers few rewards. A good game is easy to learn and offers endless rewards. *Heart of Africa*, in this sense, is a very good game.

Like *Seven Cities of Gold, Heart of Africa* is professional in every way. An excellent program, filled with surprises, the game is even more addicting than its predecessor. In *Seven Cities of Gold*, your rewards were the excitement of discovery and the favors of your monarch. *Heart of Africa* duplicates the excitement of discovery, but adds a desperate search for a lost tomb. This quest makes *Heart of Africa* an adventure as well as a simulation.

Only One World To Explore

One of the superb features of *Seven Cities of Gold* was its ability to create new worlds to explore. Players could never exhaust the game because the

ATARI 130XE

ATARI 130XE Super Computer Package
130XE Computer
1050 Disk Drive
1027 Printer
Ateriwriter +
Call for individuat & super package price

ATARI PRINTER INTERFACES

Upnnt A	54 95
Uprint AW/18K	79 95
Upnnt AW/64K	99 95
MPP 1150	59 95

INDUS GT DISK DRIVE . . . Call

ATARI 130XE SUPER PRINTER PKGS.

SG-10 Printer and U-Print A	275
Penesonic 1091 and U-Print A	309

Super Printer Peckeges heve no extre shipping cherges or credit cerd surcherges when shipped in Continentel USA

ATARI 130XE SOFTWARE

BROOERBUNO	
Print Shop	28 95
Karataka	20 95
Print Shop Graph I, II or III	19 95
Print Shop Comp	27 95
INFOCOM	
See Commodore 64 section for items end prices	
ELECTRONIC ARTS	
Archon II	21 95
Archon	16 95
Sevan Cit of Gold	21 95
Skyfox	24 95
Pinball Const	16 95
One on One	21 95
MICROPROSE	
Silant Service	23 95
Gunship	23 95
Accrojet	23 95
F-15 Strika Eagle	23 95
Kennedy Approach	23 95
OSS	
Basic XE Cart	52 95
MAC 65 XL-Cart	49 95
Action-Cart	49 95
Basic XL-Cart	39 95
All Tool Kits	20 95
BATTERIES INCLUOED	
Home Pak	34 95
Paper Clip	39 95
B-Graph	34 95
SYNAPSE	
Syncalc	32 95
Synfile	32 95
Syntrend	25 95
Syncalc Tamplatas	16 95
Loderunner Resçua	20 95
Mindwheel	27 95
Essex	27 95
Brimstone	27 95
SSI	
See Commodore 64 section for items and prices	
MISCELLANEOUS 130XE	
Hacker	19 95
Amar Cross Ctry	19 95
Flight Simulator II	34 95
Ultima II	37 95
Ultima III	37 95
Universe	69 95
Lettar Perfact	39 95
Data Perfect	39 95
Halley Project	27 95
Ultima I	23 95
Ultima IV	41 95
MMG Basic Comp	69 95

ATARI 520ST*

Atari 520ST-RGB Systam	Call
Atari 520ST-Monochroma Sys.	Call
SF314DS/DD 1 Megabyte Disk Driva	Call

We warranty all 520ST computers purchased from ComputAbility for ninaty days.

***Pleeee call for stock evellebility on Ateri ST products befors ordering by mell.**

MISCELLANEOUS ST	
VIP Professional	129 95
Ultima II	39 95
Perry Mason	34 95
Degas	27 95
Farenheit 451	34 95
Amazon	34 95
Hacker	29 95
The Final Word	94 95
Deja Vu	39 95
PC/Intercom	89 95
Hex	27 95
Crimson Crown	27 95
Mudpies	23 95
King's Quest II	34 95
Gato	34 95
Borrowed Time	34 95
Personal Prolog	79 95
Personal Pascal	64 95
Zoomracks	59 95
Mi-Term	54 95
Regent Word	34 95
Sundog	27 95
Regent Spell	34 95
Winnia/Pooh	19 95
Isgur Portfolio	Call
Michtron Utilities	41 95
Music Studio	Call
Silent Servica	Call
INFOCOM ST	
Deadline	34 95
Starcross	34 95
Zork I, II or III	29 95
Witness	27 95
Suspended	34 95
Planetfall	27 95
Sorcerer	29 95
Seastalker	27 95
Cutthroats	27 95
Hitchiker	27 95
Suspect	29 95
Wishbringer	27 95
Infidel	29 95
Enchanter	27 95
Spellbreaker	34 95
Mind Forever Voy	34 95

AMIGA

Call for Hardware and add-on peripherals prices

AMIGA SOFTWARE	
Hacker	29 95
Mindshadow	29 95
Archon	27 95
VIP Professional	139 95
Maxicom	39 95
Maxidesk	49 95
Maxiplan	119 95
One on One	29 95
7 Cities/Gold	29 95
Skyfox	29 95
Marble Madness	29 95
Return/Atlantis	29 95
Archon	29 95
Deluxe Paint	59 95
Borrowed Time	29 95
Intocom	Call
Fin Cookbook	34 95

APPLE

PRACTICAL PERIPHERALS

We carry the complete line of products
Cell for Current Prices

APPLE PRINTER INTERFACES AND BOARDS

Apricorn Parallel w/Graphics	69 95
Apricorn 16K Expansion Board	82 95
Apricorn 80 Column Board	64 95
Apricorn RS232 Interface	69 95
U-Print-Apple IIC w/64K	109 95
U-Print-Apple IIC w/16K	89 95

APPLE SOFTWARE

BROOERBUNO	
Print Shop	33 95
Print Shop Graphics I II or III	17 95
Print Shop Comp	27 95
Karateka	21 95
Carmen Sandiego	25 95
Science Tool Kit	39 95
Bank Street Writer	44 95
Fantavision	27 95
ELECTRONIC ARTS	
Adventure Const	34 95
Archon II	27 95
Bard s Tale	29 95
Imagic Football	24 95
Auto-Duel	34 95
Skyfox	27 95
Lords ot Conquest	27 95
One on One	27 95
Ultima III	39 95
Ultima IV	39 95
Moebius	39 95
EPYX	
Ballblazer	24 95
Winter Games	24 95
Summar Games II	24 95
World's Great/ Football	24 95
Eidolon	24 95
Apshai Trilogy	24 95
Koronis Rift	24 95
INFOCOM	
See Atari 520ST section for items and prices	
MICROPROSE	
See Atari 130XE section for items and prices	
MINOSCAPE	
Color Me	20 95
Crossword Magic	34 95
Halley Project	29 95
A View To Kill	27 95
Racter	29 95
The Mist	27 95
Perfect Score	49 95
Voodoo Island	27 95
Rambo	27 95
American Challenge	Call
Goldfinger	Call

SIMON & SCHUSTER	
Typing Tutor III	34 95
Kobiashi Adv	29 95
Webster Spell Chk	39 95
Webster Thesaurus	84 95
Lovejoy SAT	49 95
SIR-TECH	
Wizardry/Oiam	23 95
Wizardry/Legacy	27 95
Wizardry/Proving	33 95
Wizardry/Wernda	29 95
Wiziprint	19 95
SSI	
See Commodore 64 section for items & prices	
APPLE MISCELLANEOUS	
Beachead	23 95
Beachead II	23 95
Camemaker	27 95
Hacker	27 95
Hardball	24 95
Sundog	27 95
Star League Base	23 95
Microleague Base	29 95
PFS Software	Call
Newsroom	39 95
Clip Art Vol I	20 95
Gato	27 95
Superbase	69 95
Comp Gardening	39 95
Kings Quest II	34 95
Dambusters	27 95
Alternate Reality	27 95
Prince	47 95
Movie Maker	39 95
Zorro	27 95
Goonies	27 95
Commando	25.95
Kung Fu Master	25 95
Karata Champ	25 95
Hulk	24 95
The Hobbit	24 95
Stnp Poker	23 95
Pixit	34 95
Blazzing Peddles	34 95
Beagle Bros	Call
Math Blaster	34 95
Fight Night	27 95

We stock hundreds of programs for the Apple, Atari, C-64 and IBM. If you don't see it listed here, don't hesitate to call.

NO SURCHARGE FOR MASTERCARD AND VISA

—EST. 1982—

COMMODORE 128

C-128 Computer	Cell
1571 Disk Drive	Cell
1902 Monitor	Cell
1670 Modem	Cell

In Stock Now

IBM PC

IBM PC SYSTEMS

Contigured to your specific needs
Cell for lowest price on IBM-PC, IBM-XT or IBM-AT

Corona PC-400 Compatible	Call
Corona Portable PC Compalible	Call

PC Multifunction Bosrds

We carry the complete line of **AST, Hercules, Persdise, STB, snd Qusdrsm**

Call for current prices

IBM PC SOFTWARE

Print Shop	39 95
Print Shop Graph I	27 95
Bank Street Writer	49 95
Ancient Art ot War	29 95
BORLANO	
Sidekick	37 95
Turbo Pascal	49 95
BLUE CHIP	
Baron	34 95
Squire	34 95
Millionaira	34 95
Tycoon	34 95
OIGITAL RESEARCH	
Call for itams and prices	
INFOCOM	
See Atari 520ST for items and prices	
LEAOING EOGE	
Nutshell	69 95
LE/WP Basic	67 95
LE/Word Proc +Speller	169 95
MICROPROSE	
F-15 Strike Eagle	23 95
Kennedy Approach	27 95
Acrojet	27 95
Silent Service	27 95
MICROSOFT	
Flight Simulator	38 95
Word	249 00
Multiplan	134 95
MINOSCAPE	
Saa Appla Section for items and prices	
SIERRA	
King's Ouest	34 95
King's Quast II	34 95
Ultima II	39 95
THOUGHTWARE	
Call for items and pnces	
IBM MISCELLANEOUS	
PFS	Call
Gato	27 95
Wizardry	39 95
Strip Poker	27 95
Electric Desk	204 95
D-Base III	Call
Sideways	39 95
Home Pak	34 95
Sargon III	34 95
Peachtree	Call
Jet	34 95
BPI Business	Call
Newsroom	39 95

COMMODORE 128 SOFTWARE

Multiplan 64/128	44 95
Consultent	52 95
Paper Clip +Spell	54 95
Swiftcalc	49 95
Wordwriter	49 95
Data Manager	49 95
Fleet System II	44 95
Superbase 128	69 95
Mach V/128	34 95

COMMODORE 64

C-64 Computer	Call
1541 Disk Drive	Call
1660 Modem	49 95
1670 Modem	189

C-64 SUPER PRINTER PKGS.

SG-10 & Xetec Supergraph	279
Panasonic 1091 & Xetec Supergraph	315
Legend 1080 & Xetec Supergraph	269

Super Pnnter Packeges have no edded shipping or charge cerd surcherges when shipped in Continentel USA

B.I. 80 Column Card . . . 59.95
Whila Suppliaa Laat

Cerdco G-Wiz . . . 54.95

GENERAL HARDWARE

star micronics inc
THE POWER BEHIND THE PRINTED WORD

SG-10	**209**
SG-15	369
SD-10	339
SD-15	449
SR-10	489
SR-15	Call
PRINTERS	
Panasonic 1091	245
Legend 808	169
Legend 1080	209
Powertype	309
Juki 5510	389
Epson	Call
Panasonic 1080	215
PRINTER BUFFERS	
Microtazer	From 169
U-Butt 16K	79 95
U-Buff 64K	99 95
MOOEMS	
US Robotics 2400	469
Volksmodem 1200	189
Prometheus 1200	319
Password 1200	209
Novation	Call
PC Modems	Call
MONITORS	
Commodore 1902	Call
Commodore 1802	169
Sakata SC-100	169
Samsung 12" Green	79 95
Samsung 12" Amb	79 95
Taxan	Call
Amdek	Call

COMMODORE 64 SOFTWARE

ACCESS	
Beach-Head	21 95
Beach Head II	24 95
Raid/Moscow	24 95
Mach V-Cart	21 95
INFOCOM	
Zork I	24 95
Zork II or III	27 95
Deadline	29 95
Starcross	29 95
Witness	29 95
Planetfall	24 95
Hitchiker	24 95
Enchanter	24 95
Cutthroats	24 95
Sorcerer	29 95
Spellbreaker	29 95
SSI	
Battalion Commander	24 95
Battle ot Antietnam	32 95
Fighter Command (No Atari)	37 95
Norway 85 (No Atari)	21 95
Panzer Grenidier	24 95
USAAF	37 95
Breakthrough/ Ardennes	37 95
Kampgruppe	37 95
Phantasie (No Atari)	24 95
Broadsides	24 95
Carrier Force	37 95
Comp Ambush	37 95
Mech Brigade (No Atari)	37 95
Field of Fire (No Apple)	24 95
Op Mkt Gardan	32.95
Pro Tour Golt (No Atari)	24 95
Gemstone Warrior	21 95
Imp Galactum	24 95
Computer Baseball	24 95
Comp Quarterback	24 95

ELECTRONIC ARTS	
Adv Construction	29 95
Mail Order Monster	24 95
Racing Destruction	24 95
Ultima III	37 95
Ultima IV	41 95
Bard s Tale	27 95
See Atari 130XE section for rast of items & prices	
EPYX	
Eidolon	24 95
Koronis Rift	24 95
Winter Games	24 95
Apshai Trilogy	24 95
Fast Load-Cart	24 95
Program/Tool Kit	29 95
MICROPROSE	
See Atari 130XE section for items and prices.	
MISCELLANEOUS COMMOODRE 64	
Print Shop	28 95
Cal-Kit	34 95
Superbase 64	47 95
Karateka	20 95
Hacker	20 95
Gamamakar	27 95
Ultima II	37 95
Karata Champ	25 95
Essex	27 95
Kung Fu/ Kung Fu Master	25 95
9 Princes ot Amber	22 95
Spy vs Spy Vol II	23 95
Maxi Golf	24 95
Internat I Hockey	19 95
The Fourth Prolo	23 95
Blazing Paddles	24 95
Mirage Word	34 95
Mirage Database	34 95
Walcome Aboard	19 95
Super Huay	14 95
Spell It	34 95
Math Blastar	34 95
Word Attack	34 95
Odesta Chess	49 95
Brimstone	27 95

P.O. Box 17882, Milwaukee, WI 53217
ORDER LINES OPEN
Mon-Fri 11 a.m. - 7 p.m. CST • Sat 12 p.m. - 5 p.m. CST

ComputAbility™

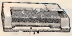
To Order Cell Toll Free
800-558-0003

For Technicel Info., Order Inquiriee, or for Wisc. Orders
414-351-2007

ORDERING INFORMATION: Plaasa **specify system.** For fast delivery send cashier's chack or money order. Persoeal and company checks allow 14 business days to claar. School P.O.'s welcoma. **C.O.D. charges are $3.00.** In Continental U.S.A. include $3.00 tor sottwara orders. 4% shipping tor hardware, minimum $4.00. Master Card and Visa orders please include card #, expiration date and signatura. WI residents please include 5% sales tax. HI, AK, FPO, APO, Puarto Rico and Canadian orders, please add 5% shipping, **minimum $5.00.** All other **toreign ardars add 15% shipping, minimum $10.00.** All orders shipped outside tha Continantal U.S.A. are shipped tirst class insured U.S. mail. It toraign shipping charges exceed the minimum amount, you will be charged the additional amount to get your packaga to you quickly and safaly. All goods ara naw and include factory warranty. Qua to our low prices all salas ara final. **All delective returos must have a return autharization number.** Please call (414) 351-2007 to obtain an R.A.# or your raturn will not ba accaptad. Pricas and availability subject to change without notica.

Lyco Computer Marketing & Consultants

SAVE ON THESE IN STOCK PRINTERS



1091 . . . . . $228

CARDCO

32K BUFFER (C-64)	59

SEIKOSHA

SP-1000UC (C-64)	169
SP-100CA Centronics	185

CORONA

LP300 Laser Printer	:2686
200361 Toner Cartridge	89

C. ITOH

Prowriter 8510 sp+	Call
15505 sp+	Call
Printmaster	Call

EPSON

LX80	209
FX85	333
JX80	Call
Homewriter 10	193
DX10	207
DX20	297
DX35	597
AP-80	244
HI-80	355
HS-80	298
FX-286 (NEW)	489
LO-800 (NEW)	529

TOSHIBA

P1340	469
P351+	1149
P341P	969
P341S	999

CITIZEN

MSP-10	255
MSP-15	355
MSP-20	337
MSP-25	495

OKIDATA

Okimate 10	179
Okimate 20	CALL
182	214
192	348
193	563
84P	645

BROTHER

HR-15XL-P	359
HR-15XL-S	359
HR-35P	839
HR-35S	839
2024L P	949

JUKI

Juki 6100	344
RS232 Serial Board	55
6100 Tractor	119
6100 Sheet Feeder	209
Juki 6300	757

LEGEND

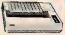
880	188
1080	Call
1380	258
1385	289
808	148

DIGITAL DEVICES

16K BUFFER	75
32K BUFFER	89
64K BUFFER	125

DIABLO

D25	549
630 API	1599
630 ECS	1759
D 80 1F	[illegible]95
P 32 CQ1	699
P 38	1749
C 150	939
DX 35 (NEW)	CALL



SG-10 . . . . $205

PANASONIC

109	228
3131 (NEW)	264
1092	325
3151	409
1080 (NEW)	209
1592 (NEW)	439

SILVER REED

EXP400	249
EXP500	295
EXP550	
EXP770	749

STAR MICRONICS

SG-10	205
SG-10C	219
SG-15	367
SD-10	319
SD-15	438
SR-10	469
SR-15	578
SB-10	589

DUST COVERS

Atari

520ST	11.95
130XE	6.99
800XL	6.99
1050	6.99
1025	7.99

Commodore

C128	7.99
1571/1541	6.99
1902	10.95
1702	8.99
C64/Vic20	6.99

Panasonic

1090/1091	8.99
1092	8.99
1093	9.99

Star Micronics

SG/SD10	8.99
SG/SD15	9.99
SR10	9.99
SR15	9.99

Okidata

82/92	8.99
83/93	9.99
193	9.99

DRIVES

INDUS

Indus GT Atari	195
GT Commodore	209

MSD

S01 Drive (C 64)	229
S02 Drive (C 64)	469

TANDON

*320K 5 ¼" Drive	115

TYMAC

MDD-640 3½" Apple Drive 640K	289

PRINTER INTERFACING

AXIOM

AT846 (Atari)	65
Parallax (C-64)	65

MICROBITS

MPP-1150 (Atari)	54
MPP-1150XL (Atari)	59
MicroPrint (Atari)	39
64K RAM Board (600xl)	65

DIGITAL DEVICES

ApeFace XLP (Atari)	49
ApeFace 12XLP (1200xl)	49
U-Print A (Atari)	54
U-Print C (C-64)	49
U-Print A16K Buffer	74
U-Print A64K Buffer	99
U-Print AP16K (Apple)	99
U-Print AP64K (Apple)	139
U-CALL RS232 (Atari)	37

MICROTEK

Dumpling GX (Apple)	59
Dumpling 16K (Apple)	89
RV-611C (Apple)	49

TYMAC

Connection (C-64)	55
Tackler (Apple)	49
PPC-100 (Apple)	39

ORANGE MICRO

GRAPPLER + (Apple)	85
Grappler 16K (Apple)	149
ORANGE (Apple)	59
Grappler CD (C-64)	79

CARDCO

G-Wiz (C-64)	54
C/?+G (C-64)	45
C/?PS (C-64)	49
C/?B (C-64)	39
C/?AT (Atari)	49
C/?AP (Apple)	49

DISKETTES

DENNISON

ELEPHANT 5¼" SSSD	11.99
ELEPHANT 5¼" SSDD	12.99
ELEPHANT 5¼" DSDD	14.99
PREMIUM 5¼" SSDD	13.99
PREMIUM 5¼" DSDD	15.99

SUNKYONG

SKC 5¼" SSDD	11.99
SKC 5¼" DSDD	13.99

MAXELL

5¼" MD1	13.99

VERBATIM

5¼" SSDD	13.99
5¼" DSDD	19.99

BONUS

5¼" SSDD	8.99
5¼" DSDD	12.99

NO LABEL DISKETTES

NL 5¼" SSDD	10.99	(Box 10)
NL 5¼" DSDD	15.99	(Box 10)

*Free Diskette Writer Pen!
*Free Storage Case!

IBM-PC SOFT-WARE

*LOTUS

Lotus 1-2-3	309.00
Symphony	439.00

*QUADRAM

Quad Jr Exp Chassis	519.00
Quad Jr Exp Memory	209.00
Quad Memory Jr	209.00

ITT XTRA XP Personal Computer!

*256K 2 Drive System	CALL
*256K 10 Meg Hard System	Call

IBM-PC COMPATABLE

*LEADING EDGE

Nutshell	69.95
LEWP Basic	65.00
LEWP Merge Print	99.00
LE Spell Correction	169.00

SSI (IBM)

Battle for Normandy	24.95
Knights of Desert	24.95
Tigers in Snow	24.95
Computer Baseball	24.95
Epidemic	22.95
Warp Factor	24.95
Cartels & Cutthroats	24.95

SYNAPSE (IBM)

Synstock	64.95
Essex	28.95
Wizard of Wall St	28.95
Brimstone	28.95

SUB LOGIC (IBM)

Jet Simulator	34.95

BRODERBUND (IBM)

Bank St. Writer	48.95
The Print Shop	34.95
Graphics Library I	22.95
Lode Runner	22.95
Ancient Art of War	27.95
Champ Lode Runner	22.95

MICROPROSE (IBM)

F-15 Strike Eagle	20.75
Solo Flight	20.75
Hellcat Ace	20.75
Spitfire Ace	18.75

AT&T

Satan 7300	CALL
6300	CALL

TOLL FREE 1-800-233-8760

TO ORDER

MasterCard VISA

CALL TOLL FREE 1-800-233-8760
In PA 717-494-1030
Customer Service 717-494-1670

or send order to
Lyco Computer
P.O. Box 5088
Jersey Shore, PA
17740

RISK FREE POLICY

In-stock items shipped within 24 hours of order. No deposit on C.O.D. orders. Free shipping on prepaid cash orders within the continental U.S. Volume discounts available. PA residents add sales tax. APO, FPO, and international orders add $5.00 plus 3% for priority mail service. Advertised prices show 4% discount for cash, add 4% for MasterCard or Visa. Personal checks require 4 weeks' clearance before shipping. Ask about UPS Blue and Red label shipping. All merchandise carried under manufacturer's warranty. Free catalog with order. All items subject

Lyco Computer Marketing & Consultants

ATARI | apple | COMMODORE

ATARI

800XL	CALL
130XE (NEW)	CALL
520ST (NEW)	CALL
1050 Drive	165
1010 Recorder	42
1020 Printer	55
1025 Printer	159
1027 Printer	179
850 Interface	109

ATARI

VIP Professional	74 50
ST Companion	15 95

ATARI SOFTWARE (NEW)

Codewriter	35 75
Filewriter	20 75
Reportwriter	20 75
Menuwriter	20 75
Home Integrator	19 75
Small Bus Inventory	11 75
Salesman Expenses	11 75
Accs Rec/Pay	11 75
Retail Invoice	11 75
Final Legacy	15 75
Adventure Writer	18 75

BRODERBUND (Atari)

The Print Shop	28 95
Graphics Library	18 95
Graphics Library II	19 50
Bank St Writer	42 75
Whistler's Brother	18 95
Spelunker	18 95
Stealth	18 95

INFOCOM (520ST)

Wishbringer	28 95
Zork I	28 95
Zork II	31 95
Zork III	31 95
Deadline	33 95
Starcross	33 95
Suspect	31 95
Suspended	33 95
Cutthroats	28 95
Sea Stalker	28 95
Infidel	31 95
Sorcerer	31 95
Enchanter	28 95
Witness	28 95
Planetfall	28 95
SpellBreaker	33 95

SPINNAKER

Delta Drawing Room	19 95
Cosmic Life Room	19 95
Up for Grabs Room	19 95

PERSONAL PERIPHERALS

Super Sketch C-64	29 95
Printer Utility C-64	14 99
Super Sketch Atari	25 95
Super Sketch Apple/Ibm	55 00

MICROPROSE (Atari)

Kennedy Approach	21 75
Crusade in Europe	24 75
Decision in Desert	24 75
Solo Flight	20 75
Nato Commander	20 75
Spitfire Ace	18 75
F-15 Strike Eegle	20 75
Hellcat Ace	18 75
Acrojet	21 75
Silent Service	21 75
Gunship	21.75

SUB LOGIC (Atari)

Flight Simulator II	***32.75***
Night Mission Pinbell	20 75

GLENTOP

ST Logo Book	15 95

MARK of UNICORN (520ST)

HEX	29 95
MINCE	129 95
PC/InterComm	99 95

FTL GAMES

Sundog	29 95

SIERRA

ULTIMA II	39 95

HABA

10 Megabyte	CALL
Hippo C Compiler	49 95
Check Minder	49 95
Business Letters	36 95
Wills	36 95

SOFTECHNICS

Timelink	75.00
Rhythm	39 95

SCARBOROUGH (Apple)

NET WORTH	48 75
Improved Mastertype	22 75
Mastertype Filer	22 75

BRODERBUND (Apple)

The Print Shop	31 50
Graphics Library	18 50
Graphics Library II	18 95
Paper Refill	11 95
Bank St Writer 64K	41 50
Bank St Writer 128K	42 75
Bank St Filer	42 75
Bank St Mailer	42 75
Bank St Speller	42 75
Carmen Sandiego	22 75
Karateka	22 75
Captain Goodnight	22 75
Muppey Cruise	25 75

SUB LOGIC (Apple)

Flight Simulator II	***32.75***
Night Mission Pinball	20 75

MICROPROSE (Apple)

Crusade in Europe	25 75
Decision in Desert	25 75
F-15 Strike Eagle	21.75
Nato Commander	21.75
AcroJet	21 75
Silent Service	21 75
Gunship	21 75

SSI (Apple)

Mech Brigade	36 75
Norwey 1985	22 75
Market Garden	29 75
Six Gun Shootout	24 75
50 Mission Crush	24 75
Phantesie	24 75

INNOVATIVE CONCEPTS

Flip-n-File 10	3 50
Flip-n-File 15	8 25

COMMODORE

C-128 NEW	CALL
1571 Drive	CALL
1572 Drive	CALL
1902 Monitor	CALL
1670 Modem	CALL
C-64 Computer	CALL
1541 Drive	189
MPS801 Printer	LOW
1702 Monitor	199
Simon's Basic	24 75
Assembler 64	34 75
Super Expander	22.75

MICROPROSE (C-64)

Kennedy Approach	21 75
Crusade in Europe	24 75
Decision in Desert	24 75
Solo Flight	20 75
Nato Commender	20 75
Spitfire Ace	18 75
F-15 Strike Eagle	20 75
Hellcet Ace	18 75

CARDCO

Digitizer Camera	189 95
32K Printer Buffer	59 95
Numeric Keypad	34 95
CB/5 5-slot Board(64)	54 00
CB/2 2-slot Board(64)	25 00
S'More Basic Rom	49 95
Write Now-84	35 00
Mail Now-84	29 00
Spell Now-64	29 00
File Now-64	29 00
Paint Now-64	29 00
Calc Now-64	29 00

EPYX
(C-64)

Fast Load	28 75
Breakdence	23 75
Greetest Baseball	24 75
Summer Gemes	28 75

BRODERBUND

The Print Shop	28 75
Graphics Library	18 75
Graphics Library II	19 75
Graphics Library III	19 95
Karateka	19 75
Castles Dr Creep	19 75
Bank St Writer	32 75
Loderunner	20 75
Mask of the Sun	24 75
Spelunker	19.75
Serpent's Star	24 75
Whistler's Brother	18 75

SUB LOGIC (C-64)

Flight Simulator II	***32.75***
Night Mission Pinball	20 75

SSI (C-64)

Colonial Conquest	24 75
Wings of War	24 75
Computer Ambush	34 75
Field of Fire	24 75
Fighter Command	36 75
Kampfgruppe	36 75
Mech Brigede	36 75
Market Garden	29 75
Six Gun Shootout	24 75
Computer Beseball	24.75
Computer Quarterbeck	24 75
Imperium Galactum	24 75
Phantasie	24 75
Cartels & Cutthroets	24 75
50 Mission Crush	24 75
Questron	24 75

SPINNAKER (C-64ROM)

Cosmic Life ROM	19 75
Jukebox	19.75
Alphabet Zoo	17 75
Alf in Color Caves	19 75
Up for Grabs	19 75
Delte Drewing	19 75
Kids on Keys	16.75
Kindercomp	14 75
Facemaker	17.75
Frection Fever	17 75

MODEMS

US ROBOTICS

Password 1200M	229
Password 1200F	229
Password 300M	139
Password 300F	139
Autodial 212A	359
PCM5	319
PCM64	519
PCM256	769
S-100	379
Courier	469
Microlink	469
Telpac Ms-DOS	79

TELE LEARNING

Total Telecommunications (C-64)	29 95
AP 250 (300 Baud Apple)	69 95
IB-250 (300 Baud IBM)	69 95

HAYES

Smartmodem 300	133
Smartmodem 1200	377
Smartmodem 1200B	347
Smartmodem 2400	598
Micromodem IIE	135

ANCHOR

Volksmodem	55
Volksmodem 12	186

NOVATION

IBM 300/1200 MS-DOS ext	319
IBM 300/1200 CPM-86 ext	319
IBM 300/1200/2400 ext	529
IBM 300/1200/2400 MS-DOS	579
IBM MS-DOS int	325
IBM CPM-86 int	325
Macmodem 300/1200	315

RACAL-VADIC

2400PC	549
2400PA	799
2400V	559
1200PC	329
300V	205
300PC	199

3.5" DISKETTES

DENNISON

3 5" SSDD	5 pak	14 95
3 5" SSDD	10 pak	26 95

MAXELL

3 5" SSDD	10 pak	29 95
3 5" DSDD	10 pak	36 95

3 M

3 5" SSDD	10 pak	26 95
3 5" DSDD	10 pak.	32 95

MONITORS

SAKATA

SG 1000 12" Green	99
SA 1000 12" Amber	109
SG 1500 12" Green TTL	119
SA 1500 12" Amber TTL	129
SC 100 13" Color Comp	209
SC 200 13" RGB	389
STS1 Tilt Stand	29

PRINCETON GRAPHICS

MAX-12 Amber	185
HX 12 RGB	465
SR-12 RGB	595

TAXAN

115 12" Greeen Composite	CALL
116 12" Amber Composite	CALL
121 12" Green TTL	135
122 12" Amber TTL	145
220 14" Color Composite	259
410 12" RGB Hi Res IBM	329
420 12" RGB Super Hi IBM	409
440 12" RGB Ultra Hi Res	555
Tilt Stand	35

TEKNIKA

MJ 10 Composite	179
MJ 22 RGB	255

PANASONIC

DT1300D 13" RGB/Composite	247
DTM140 14" RGB/Composite	329
DTH103 10 RGB Hi Res	395
DTS101 10 Composite	175
DT1000G 10" RGB	166
TX12H3P 12 Color	419
TR120M1PA 12" Green	109
TR120MBPA 12" Amber	109
TR122M9P 12" Green IBM	148
TR122MYP 12" Amber IBM	148

ZENITH

ZVM 122A Amber	75
ZVM 123G Green	75
ZVM 124 Amber IBM	129
ZVM 131 Color	275
ZVM 133 RGB	389
ZVM 135 Composite	449
ZVM 136 Hi Res Color	589
ZVM 1220	95
ZVM 1230	95
ZVM 1240	149

AMDEK

300 Green	118
300 Amber	128
310 Amber IBM	155
Color 300 Audio	234
Color 500 Composite	369
Color 600	397
Color 700	495
Color 710	569

NEC

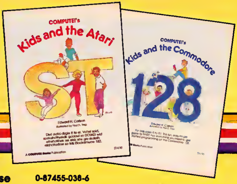
JB-1260 Green	95
JB-1201 Green	135
JC 1215 Color	235
JC 1216 RGB	375
JC 1460 Color	265

LYCO COMPUTER
AMERICA'S MAIL ORDER HEADQUARTERS!

NEW

COMPUTE! Books

For Kids



0-87455-038-6
$14.95

0-87455-032-7
$14.95

Help your children learn the basics of computer programming with these two new entertaining and educational books from COMPUTE!.

Each book contains easy-to-follow instructions, programming examples, quick reviews, and colorful illustrations. Written in COMPUTE!'s clear, easy-to-understand style, the books offer hours of entertainment while helping kids (and adults) learn to program in BASIC.

If you're acquainted with BASIC, you can easily write your own games and applications on Atari's ST or Commodore's 128 computers. Over 30 sections—all with instructor notes, lessons, assignments, and lively illustrations—entertain and amuse as you learn to use these powerful computers. *COMPUTE!'s Kids and the Atari ST* and *COMPUTE!'s Kids and the Commodore 128*, in the bestselling series from author Edward Carlson, are gentle introductions to programming your new computer. Clear writing and concise examples, both trademarks of this series, make it easy for anyone—child or adult—to learn BASIC painlessly.

**Look for these and other books from COMPUTE!
at your local book store or computer store. Or order directly from COMPUTE!.**

To order, call toll free in the US 1-800-346-6767 (in NY 212-265-8360) or mail the attached coupon with your payment to COMPUTE! Books, P.O. Box 5038, F.D.R. Station, New York, NY 10150.

Please send me the following COMPUTE! books. My payment is enclosed.

________ ***COMPUTE!'s Kids and the Commodore 128,*** (032-7) $14.95 each ________

________ ***COMPUTE!'s Kids and the Atari ST,*** (038-6) $14.95 each ________

ALL ORDERS
MUST BE
PREPAID IN
U S. FUNDS

Subtotal ________

NC residents add 4.5% sales tax ________

Shipping and handling per book ________
(in U.S. and surface mail, $2.00 per book; airmail, $5.00 per book.)

Total amount enclosed ________

□ Payment enclosed (check or money order)

□ Charge □ MasterCard □ Visa □ American Express

Account No. ________________ Exp. Date ________________
(Required)

Name ________________________________

Address ________________________________

City ________________________________

State ________________ Zip ________________

Please allow 4–6 weeks for delivery.

36412011

COMPUTE!™ Publications, Inc. abc
One of the ABC Publishing Companies
825 7th Avenue 6th Floor New York, NY 10019
Publishers of COMPUTE! COMPUTE!'s Gazette COMPUTE!'s Gazette Disk, COMPUTE! Books, and COMPUTE!'s Apple Applications

COMPUTE! books are available in the U.K., Europe, the Middle East, and Africa from Holt Saunders, Ltd., 1 St. Anne's Road, Eastbourne, East Sussex BN21 3UN, England and in Canada from Holt, Rinehart, & Winston, 55 Homer Avenue, Toronto, ON M8Z 4X6.

program could make the world different each time. Surprisingly, *Heart of Africa* offers no such option. There are very good reasons for this—the time limit, and the quest itself—but perhaps the game would be even more complete if each Africa could be a new one. Discoveries are less exciting when you know about them beforehand. Furthermore, the desperate feeling of being hopelessly lost, which *Seven Cities of Gold* presented so well, cannot happen here. If the game has a flaw, this is it.

But the flaw is easily overcome. The romance of uncovering the Dark Continent captures the imagination today as much as ever, perhaps because there remain no large, unexplored land masses anywhere in the world. *Heart of Africa* lets you canoe down the Congo, meet a Zulu chief, and even get caught in a whirlpool near Stanley Falls. Khartoum, Timbuktu, the Zambesi, Lake Tanganyika, Tangier—they're all there, waiting for you, ready to throw you many surprises.

An almost flawless development of an already excellent game system, *Heart of Africa* should excite anyone who found *Seven Cities of Gold* even remotely interesting. Now, if only I could find Dr. Livingstone.

Heart of Africa
Electronic Arts
2755 Campus Drive
San Mateo, CA 94403
$32.95

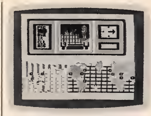
Hacker

Todd Heimarck, Assistant Editor

Requirements: Commodore 64 or 128; Apple II-series computer with at least 64K RAM; Atari 400/800, XL, or XE with at least 48K RAM; IBM PC/PCjr with at least 128K RAM; Atari ST; Amiga; or Apple Macintosh. Disk only.

When the first thing you see is the prompt LOGON PLEASE:, you want to reach for the rule book. But apart from a card that tells you how to load and run this game, there are no instructions. None at all.

The premise behind *Hacker* is that you have stumbled across a telecommunications system about which you know nothing. Being a good hacker (if that's not an oxymoron), you feel the urge to break in and explore. Try a few passwords; unless you're very lucky, none of them will work. After several failures, the system logs you off and the game ends. Or does it? Some random characters appear on the screen, and the computer indicates that a security malfunction has occurred. You're in.



On the trail of corporate skullduggery in Activision's Hacker *(Commodore 64 version).*

The logon sequence is very realistic. Once, at the beginning of a game, someone walked into the room and watched me guess at a few passwords. Hearing that we were trying to get into an unknown system, which might be a government computer, and then seeing the security malfunction message, he got worried and reminded us that it's illegal to do what we were doing. That's the great appeal of *Hacker*, the feeling that you're doing something wrong and that you might get caught. Who knows, the FBI might even show up at your door and confiscate your computer.

After you enter the system, the game becomes less realistic On an actual telecommunications system, everything would be straight text. The author of *Hacker*, in the interests of playability, has inserted some high-resolution graphics—unlike anything you'd see on a true bulletin board system or information service. However, the graphics do add a lot to the game.

Remote-Control Robots

You soon discover that you've come across a company involved in some sort of top-secret illegal project. This makes you feel less guilty about breaking into someone's system; you can seek out more information about this project and bring the culprits to justice.

The company owns a vast network of subterranean tunnels, and their computer (to which you've gained access) controls robots that travel through the tunnels. By using the robot to explore the tunnel network and occasionally coming to the surface, you can accumulate more details about the project. I'll say no more about the most effective techniques for winning because an important part of the game is figuring out what's going on.

The game play is almost identical in the versions I tried on the Commodore 64, Atari 520ST, and Amiga. The newer 16-bit machines (Amiga and ST) displayed slightly better graphics than the 64 because their screens have higher resolution and more colors. The ST version works on both monochrome and color monitors.

All things considered, *Hacker* is a worthy addition to your software collection, especially if you enjoy adventure games that require a bit of thought and an investment of time.

Hacker
Activision, Inc.
2350 Bayshore Frontage Road
Mountain View, CA 94043
$24.95 (Atari 400/800/XL/XE)
$29.95 (Commodore 64/128)
$39.95 (Apple and IBM)
$44.95 (ST, Amiga, and Macintosh)

MasterType's Writer For Apple

Stephen Levy, Book Editor

Requirements: Apple IIc or Apple IIe with 128K RAM and a printer. A Commodore 64/128 version is scheduled for release this spring.

Does the world really need another word processor? After all, *MasterType's Writer* does all the things most word processors do. Using direct commands or on-screen menus, you can write, edit, save, search, move, change, find and replace, and print just as you can with most full-featured word processing programs.

So what makes *MasterType's Writer* special? If you're using it with an Apple IIc/IIe and an Imagewriter or Imagewriter II printer, and if you need multiple fonts—including some very large print styles—*Writer* is worth a closer look, even if you already have a word processor. With an Imagewriter or Imagewriter II, *Writer* can dump an exact copy of what's on the screen to the printer. *Writer* comes with eight fonts which can be loaded from disk and saved with your text. Among the styles are fonts that print very large type to the screen, quite suitable for use by young children just learning to read; proportionally spaced fonts of various sizes; and a style that is quite suitable for use on a monochrome monitor.

Each font can be edited, so you can modify those provided or design your own completely new font. And once created, you can use the screen dump feature to duplicate text written with the new font on paper.

If you have an Imagewriter II with a color ribbon, it's a simple matter to print text in color—simply underline the text to be printed in green with a green line, blue text with the blue line, and so on. Again, you get an exact copy on paper.

Some Nice Touches

In addition to the fancy printing features, *MasterType's Writer* includes a few other extras. For example, the on-disk tutorial is well done and is a good introduction to using the program. Many people will return to the tutorial a second or third time even after they've started creating documents.

With *Writer*'s dual windows, you can work on two documents at the same time. You can have an outline in one window and the text you're writing in the other. If you've never used this kind of feature before, you might not miss it; but once you've tried it, you'll wonder how you got along without it. *Writer*'s dual windows have the added advantage of allowing you to decide how much of the screen each window will occupy at any time.

The manual is arranged in alphabetical order with entries for most of the terms you're likely to look up. Usually a term refers you to the appropriate instructions. If you're the type who likes to jump right in, you may find the manual a bit frustrating. But if you've tried the on-disk tutorial, you'll find the manual easy to use. And once you've been using *Writer* for awhile, an alphabetically arranged manual makes locating information a snap.

Another powerful feature of *MasterType's Writer* is keyboard macros—you can recall a series of instructions with one or two keystrokes. Macros are especially handy for storing a series of often-used words. If you're writing a book report, for example, you might need to type the author's name or the book's title many times throughout the report. By defining these phrases as macros, you can type them simply by pressing two keys.

Since macros can include program commands as well as ordinary characters, you can create macros for such purposes as saving your document on disk. Then, whenever you want to save the current copy of your work, you just press two keys.

Ease Of Use

MasterType's Writer gives you the choice of using direct commands—usually accessed by pressing CONTROL and one other key—or menus. Moving through the menus is easy and fast and saves you the trouble of memorizing commands. The menus are ideal for those new to word processing. Direct commands are faster for some functions, but for others save little more than one or two keystrokes. Most people will probably use a combination of both menus and direct commands.

If you revise text often, one aspect of *MasterType's Writer* you may find annoying is its text entry and editing line. *Writer* doesn't allow full-screen editing; all text must be entered and edited on the bottom line of the current window. That means you must press the cursor keys to move the line you wish to edit to the bottom of the window. This isn't a problem when first entering text, but later, when editing, you can't see what comes immediately after the line you're trying to alter without continuously moving the text up and down.

For whom is *MasterType's Writer* most suitable? It should be strongly considered by those who have never used a word processor, teachers or students who plan to use it in schools, Apple users with an Imagewriter printer, or anyone who is unhappy with their current word processing program.

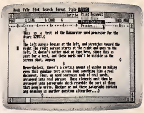
MasterType's Writer
Scarborough Systems, Inc.
55 South Broadway
Tarrytown, NY 10591
$69.95

HabaWriter For The Atari ST

George Miller
Assistant Technical Editor

Requirements: Atari ST with at least 512K RAM and a compatible printer.

If you've been using *ST Writer*, the free word processor from Atari, but have been wishing for a program that supports the drop-down menus and windows of GEM, then *HabaWriter* is for you.



This is a test of the Habawriter word processor for the Atari 520ST.

The left margin begins at the left, and stretches toward the right The right margin starts at the right and goes to the left. It doesn't matter what we type here, since this is just for a test, and these words won't be visible in the screen shot, anyway

Nevertheless, there's a certain amount of wisdom in making even this mundane test screen look something like a real document. Thus, we need sentences made of real words, arranged into real phrases These elements must then be arranged into paragraphs which resemble the sort of thing that people write. Whether or not these paragraphs contain any meaning is another question altogether....

HabaWriter takes advantage of the GEM environment and includes all the features we've come to expect in a good word processor. Even more important, *HabaWriter* is easy to use. The instruction manual isn't very long—only 46 pages. If you're accustomed to other word processors, this may seem strange. Many programs have entire books devoted to their use, and sometimes it's necessary to enroll in special classes to become really proficient. Even though the size of the *HabaWriter* manual is small, all the information you'll need is there.

When you start up *HabaWriter*, you see a menu bar at the top of the screen with seven headings: Desk, File, Edit, Search, Format, Style, and Print. Just as on the GEM desktop, each menu instantly drops down when you point to it with the mouse. A click of the mouse button picks any selection on the current menu. Happily, *HabaWriter* is an intuitive program—the way you want to do something is probably the way *HabaWriter* requires you to do it. And the commands on the menus allow you to do just about everything you can think of.

If you prefer not to use the mouse, the ST's ten special function keys let you access most of *HabaWriter*'s features. If you can't remember which key

THE EPSON LIBRARY FROM
MERDYNE PUBLISHERS

NEW:

EPSON
CONNECTION:
ATARI

It's time to use your ATARI 800XL for more than playing games. Manage your household, succeed in school, do reports for work. This book shows you how. (L.E. Zeitz . . $16.95)

ALSO NEW

EPSON Printers: Tips & Secrets
(Darnall & Corner . . . $16 95)
EPSON Guide to PC Communications
(Banse $16 95)

LOOK FOR THE EPSON LIBRARY AT YOUR EPSON DEALER, LOCAL BOOKSTORE OR YOU CAN ORDER FROM

MERDYNE
Merdyne Publishers. Inc.
184 Fifth Ave., New York, NY 10010
212/255-8448. TELEX MERDYN

does what, just press the Help key. A window opens on the screen to display a convenient chart of the function key commands.

You can even use the Style menu to change the on-screen typeface from plain text to boldface or underline. Underlined text, however, is displayed with true underlining only on the monochrome monitor in high-resolution mode; it won't be underlined on the color monitor in medium resolution. But when you print your document, the text *is* underlined. (If you're using a printer that's capable of underlining, of course.)

Multiple Windows

HabaWriter lets you use the mouse for most functions that would require special commands with other word processors. For instance, you can define a block of text simply by dragging the mouse cursor over it. To delete the block, you would then select the Cut option from the Edit menu. Even though the text is erased off the screen, it's still kept temporarily in a clipboard and may be pasted back into the document wherever you wish.

HabaWriter uses screen windows for other purposes, too. You can open up to six windows to display more than one document at a time. Moving text between the windows is as easy as moving text within a document. And the size of the windows can be changed, just as with any GEM window. Files can even be combined, allowing you to work with longer documents by using the Paste Document option.

On-screen rulers let you easily set tab stops anywhere you want, and horizontal scrolling lets you create documents as wide as 132 columns. (Only 80 columns are displayed on the screen at a time.)

Using options on the Format menu, you can center text on the screen, align it to the right or left margins, or "justify" the text, just as professionally typeset pages appear.

HabaWriter's use of the GEM environment and its wide range of features make it one of the most attractive application programs to date for the Atari ST. You'll find it's a snap to give your correspondence and club newsletters a polished look—without much of the strain that's usually involved in learning how to use a new word processor.

HabaWriter
HabaSystems, Inc.
6711 Valjean Avenue
Van Nuys, CA 91406
$74.95

©

MUST LIQUIDATE!
COMMODORE PLUS/4™
COMPUTER SYSTEM
AT BELOW DEALER COST!

NOTE: Monitor not available. The Plus/4™ can hook up to your TV.
FREE! WITH PURCHASE OF COMPLETE SYSTEM 20 TOP GRADE FLOPPY DISKS RETAIL VALUE: $109.00
Ideal for home or business! Perfect for programmers!

Factory new!

Factory warranted by Commodore®. Built-in software for word processing, data processing, spreadsheets and 128 color graphics!

Commodore® designed this Plus/4™ specifically for **programmers and small businesses!** And then they made it **VERY EASY** to learn and use for novices. Popular business software is available for a variety of purposes. For programmers, this machine has easy-to-use powerful commands and 60K of usable memory. Can hook up as many as four disk drives.

Team up the computer with our compatible units of famous brand, factory reconditioned and warranted **DISK DRIVE** and **DOT MATRIX PRINTER!** Sorry, we can't print the brand name of the disk drive and printer. But phone us Toll-Free and we can tell you. With Disk Drive for data storage and Printer for "hard copies", you'll have a complete system...at low liquidation prices. Order **TODAY!**

Units sold individually or in any combination you dasire	Original List Prica	YOUR COST
PLUS/4 COMPUTER Item H-1087-5035-001 Ship, hend $8 00	$299.00	$79
DISK DRIVE Item H-1087-3553 013 Ship, hand $8 00	$269.00	$149
PRINTER Item H-1087-3831-005 Ship, hend $7 00	$200.00	$119
TOTAL Ship, hand $23 00	$768.00	$347

Additional Features of
COMMODORE PLUS/4
Data base of 999 records. Computar holds 99 lines of text before it must be transferred to disk drive. Excallent terminel for use with modem. 128 colors available for graphics. Split screen and windowing capabilitias. Compatible with ell Commodore hardware excapt joystick and dataset. NOT compatibla with C64 software

DISK DRIVE
Intelligent, high-speed. External 5¼" floppy diskette recordar. 2K RAM, 16K ROM. Maximum storaga of 170K formatted data. 35 tracks. Uses single sided, single dansity disk. Seriel interface. Second serial port for chaining second drive or printer. Data transfer rete of 400 bytes per second

DOT MATRIX PRINTER
Bi-directional 6 x 7 dot matrix impect printar. 60 charactera per aecond. Hae uppar and lower case letters, numerals and symbols. All PET graphic characters. Standerd friction feed. Meximum of 80 columns width, dot eddressabla. CBM, ASCII cheracter codes. Original plus maximum of two copias. Paper width. 4.5" to 8 5" Size: 13" W x 8" D x 3¼" H. Weight 6½ lbs

COMPLETE SYSTEM
WITH 20 FREE FLOPPY DISKS
Total Original List . . . $877.00
Liquidation Priced At Only . . . $339
Item H-1087-5035-019 Ship, hand $19 00

Credit card customers can order by phone, 24 hours a day, 7 days a week
MasterCard VISA

Toll-Free: 1-800-328-0609

Your check is welcome!
No deleys in orders paid by check.

Sales outside continental U S are subject to special conditions. Please call or write to inquire

C.O.M.B. Direct Merketing Corp.
Authorized Liquidator
14605 28th Avanue North
Minneapolis, Minnesota 55441-3397

C O M B Direct Marketing Corp — Item H-1087
14605 28th Ave N./Minneapolis, MN 55441-3397
Send the following items (Minnesota residents add 6% sales tax. Allow 3-4 weeks for delivery. Sorry, no C O D orders.)
Send___COMPLETE SYSTEM(s) Commodore® Plus/4™, Disk Drive, Printer and Floppy Disks Item H-1087-5035-019 at $339 each plus $19 each for ship, handling
Send___COMMODORE® PLUS/4™ COMPUTER(s) Item H-1087-5035-001 at $79 each plus $8 each for ship, handling
Send___DISK DRIVE(s) Item H-1087-3553-013 at $149 each plus $8 each for ship, handling
Send___PRINTER(s) Item H-1087-3831-005 at $119 each plus $7 each for ship, handling
□ My check or money order is enclosed. (No deleys in processing orders paid by check, thanks to TeleCheck.)
Charge □ MasterCard® □ VISA®
Acct No ________ Exp. ___/___
PLEASE PRINT CLEARLY
Name ________
Address ________
City ________
State ________ ZIP ________
Phone (____) ________
Sign Here ________

Loading And Linking Commodore Programs
Part 2

Jim Butterfield, Associate Editor

Are you running out of memory for your programs? You don't necessarily have to buy a bigger computer. This month's installment shows how a technique called chaining *lets you break up a large program into smaller parts to work on a common task. The technique applies to all Commodore computers, with either disk or tape.*

There are three major ways of connecting programs together. *Chaining* allows several. programs to perform a job, each program continuing the work that a previous program has started. *Load linking* lets one program load another program, with the new program starting fresh on a new task. *Overlaying* lets a main program call in additional subroutines, data tables, or graphics information. This month we'll discuss chaining.

When one of a series of programs has completed its share of the work, it may chain to a following program to continue processing the data. In effect, several programs group together to create a bigger program. On Commodore computers, chaining works with disk or tape. It's more common with disk because the various programs can be brought in more quickly. If used with tape, you can arrange the programs sequentially on the cassette so little time is lost in searching for the next program. We'll use disk for the following examples, but they can be readily converted to tape.

Why Chaining?

The most obvious reason to chain programs is to save memory space. On small computers, there isn't enough room for big jobs. So the program is broken up into "chunks." Each chunk is small enough to fit into memory, each does a specific task, and together they do the whole job. Even on computers that seem to have lots of memory, you may need to resort to chaining to relieve congestion. For instance, even though the Commodore 64 begins with 38,911 bytes of free memory, arrays of data can quickly fill up much of this work area.

Sometimes program flow is an important reason for chaining. If a statistical program has been processing some data, it might ask the user to choose from several options (draw a graph, print the data, etc.). Depending on which option is chosen, it may be convenient to call in a selected program to do the next job. In this way, the original program needn't be cluttered with code to cover all the possible options; instead, the options are handled by programs called in as required.

Likewise, it's possible to write a program that starts up in several different ways. In one case, it might collect the data it needs from DATA statements. Another time, it might require input from the keyboard. On still other occasions, it might compute the data, read it from a file, or detect it by external sensors. No matter. We'll start up whatever "acquisition" program is appropriate, and when the data is ready to be processed, the computer can chain to a common processing program.

Chaining is also a worthwhile exercise which can force you to break your programs into well thought-out modules. Your program can't leap about at will, since it can only reach whatever is in the current module; and you must tie up loose ends before you go to the next unit. Each time you chain, FOR-NEXT loops are scrapped, subroutine RETURNs are canceled, and the DATA pointer is RESTOREd. You must make sure that these program areas are tidy before you chain, since they will otherwise be lost.

Program Architecture

A major advantage of chaining is that *you don't lose variables between programs*. Values, strings, and arrays that have been worked out by a previous program are carried through to the next program segment. This is useful, but it also calls for careful handling—we don't want to mash these values inadvertently.

Figure 1 shows how programs, variables, and arrays lie in memory. The point marked start-of-BASIC is where the program starts in memory. Behind the program is a point called start-of-variables; beyond this point the computer stores variables and arrays.

You usually don't need to know the exact addresses of these memory points; the computer takes care of the housekeeping for you. String variables go into this area, too—although not the strings themselves, just three-byte *descriptors* that say where the strings are located and how long they are. (More on this later.)

Suppose you have a large program that chains to a smaller program. Figure 2 shows this happening.

The variables don't move; behind the second program is wasted space that isn't used. This creates no problem when you run the program. However, after this kind of chaining has taken place, you should not SAVE the second program or you'll save the wasted area too (SAVE always stores from the start-of-BASIC point to just before start-of-variables).

Here comes the problem. Let's take the reverse situation: a small program that chains to a larger one. Figure 3 shows the difficulty that results.

The big program overwrites and destroys the variables created by the first, smaller program. To keep this from happening, our first program must be the biggest of the two, or at least the same size.

If several programs are chained together, this rule always applies. The first program must be as big or bigger than any other program. It sets the start-of-variables point, and it must set it high enough so that all following programs won't run into trouble (for more information on this point, see "Commodore Program Chaining," COMPUTE!, December 1985). The Commodore 128 in 128 mode doesn't need to worry about this problem. Since it keeps variables in a separate memory bank, loading a new BASIC program can't harm them.

Figure 1. BASIC program storage

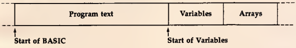


Figure 2. Chaining a smaller program from a large one.

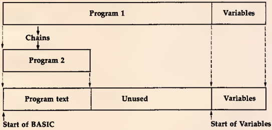
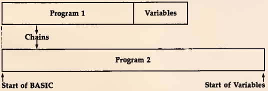


Figure 3. Chaining a larger program from a smaller one.



Chaining a longer program destroys the original program's variables.

Strings And Descriptors

As noted earlier, the variable and array area holds string information (the descriptors), but not the strings themselves. There are two places where the actual strings might be, and it's important to know about them. Say that your program contains a line like this:

```
370 A$="GORILLA"
```

When this line executes, the computer makes an entry in the variable table showing that there is now a variable called A$, that its length is seven characters, and that it is located at its present position in the program text itself. Except on the 128, the string is used from where it lies within the program. The computer decides that there's no point in making an extra copy of GORILLA; when it needs this string, it takes it from the BASIC program line. This type of string is called *static* because it never moves from its original location. Static strings can mean trouble if you chain programs: Since chaining replaces the original program text with a second program, all static strings—which exist only in the first program's text—are destroyed.

There's a second kind of string, and that's the one we must use here. If a program contains a statement like INPUT A$, the string which is typed by the user must be stored somewhere. This is called a *dynamic* string; the computer stores it in a safe place where it won't be disturbed by chaining.

Dynamic strings are created in two ways: by INPUT or GET statements and by string manipulations (LEFT$, RIGHT$, STR$, concatenation, and so on). It's simple to change a static string into a dynamic one. The statement A$="GORILLA" + "" concatenates (adds together) the strings "GORILLA" and "". Since "" is a null (empty) string, this statement really means "add nothing to the string GORILLA." Though the contents of the string don't change, the computer is convinced that we now have a new string which must be stored elsewhere in memory.

Again, the Commodore 128 in 128 mode doesn't need to worry about this problem. Strings are kept in a separate memory bank, and there's no such thing as a static string in 128 mode.

Chaining Rules

Let's summarize the rules for well-chained programs:

- The first program in the chain must be as big or bigger than all subsequent programs.
- Any strings you need to pass from program to program must be dynamic, not static.
- If you use DEF FN definitions, redefine them in each program.
- Arrays should be DIMensioned only once, preferably in the first program.

A Short Example

Let's write a small series of programs to demonstrate how this works. Our first program is called MAIN:

```
100 IF N>Ø GOTO 2ØØ
```

The variable N can only be zero when we start, so we won't jump ahead. But if we ever chain back to this program, we'll take the branch to line 200.

```
11Ø PRINT "SIMPLE GRADEBOOK DE
   MO"
12Ø DIM N$(15),M(15)
13Ø N=8
```

For simplicity, we'll assume eight students. When the program runs, you can invent their names and numeric grades.

```
14Ø FOR J=1 TO N
15Ø PRINT "STUDENT";J;
16Ø INPUT "NAME";N$(J)
17Ø INPUT "GRADE";M(J)
18Ø NEXT J
```

Running the program at this point gives you data on eight students. If you ever chain back to this original program, it will branch to line 200 (remember the IF test in line 100).

```
2ØØ PRINT
21Ø PRINT "DO YOU WANT TO--"
22Ø PRINT "1. CALCULATE AVERAG
   E"
23Ø PRINT "2. CALCULATE HIGH/L
   OW SCORES"
24Ø PRINT "3. QUIT"
25Ø PRINT
26Ø INPUT "YOUR CHOICE (1-3)";
   C
27Ø ON C GOTO 3ØØ,31Ø,32Ø
28Ø GOTO 26Ø
3ØØ LOAD "C.AVG",8
31Ø LOAD "C.HIL",8
32Ø END
```

Note that line 300 will not run into line 310, nor 310 into 320. The moment you perform LOAD within a program, the new program loads and runs immediately. Type this program and then save it as MAIN (don't save it under any other filename). Now type NEW and enter program C.AVG as follows:

```
1ØØ PRINT
11Ø A=Ø
12Ø FOR J=1 TO N
13Ø A=A+M(J)
14Ø NEXT J
15Ø PRINT "AVERAGE SCORE,";N;"
   STUDENTS=";A/N
16Ø PRINT
17Ø LOAD "MAIN",8
```

That's it. Check it closely and save it as C.AVG (again, the filename is important; don't change it). Now type NEW and enter program C.HIL as follows:

```
1ØØ PRINT
11Ø H=M(1):L=M(1)
12Ø FOR J=1 TO N
13Ø IF H<M(J) THEN H=M(J)
14Ø IF L>M(J) THEN L=M(J)
15Ø NEXT J
16Ø PRINT "HIGH SCORE:";H;" BY
   ..."
17Ø FOR J=1 TO N
18Ø IF H=M(J) THEN PRINT N$(J)
19Ø NEXT J
2ØØ PRINT "LOW SCORE:";L;" BY
   ..."
21Ø FOR J=1 TO N
22Ø IF L=M(J) THEN PRINT N$(J)
23Ø NEXT J
24Ø PRINT
25Ø LOAD "MAIN",8
```

Again, check your typing closely and save the program as C.HIL to complete the set. Now load program MAIN and you're ready to try out chaining. Note that MAIN is definitely larger than the other two. If there's any doubt in your mind, add some extra REM statements to MAIN to make it bigger.

Side Effects

We mentioned earlier that the act of chaining causes certain things to happen. FOR-NEXT loops are scrapped, subroutine RETURNs are canceled, and the DATA pointer is RESTOREd. That makes sense: You can't RETURN to a program that has disappeared, for example. And occasionally, these side effects can be useful. For instance, can a program ever chain to itself? The answer is yes, but at first it's hard to see why you'd want to do so. What's the point of loading a program that's already there? The answer lies in these side effects.

Sometimes a program gets stuck deep in a subroutine and can't find its way out. With good programming, this should never happen. All subroutines should RETURN neatly, and if there's an error or similar anomaly, the information should be logged into a flag and detected at the appropriate program level. It's easy to give that sort of advice—but sometimes a program is deep within several nested levels of subroutines when the user commands, "Forget all this and take me back to the menu." Sensible programmers know that you can't jump directly out of these subroutines back to the main menu, and it's a long, long trail to backtrack the whole way.

In case of emergency, you can chain the program to itself. As it loads itself back in, it shakes off all the FOR-NEXT loops and subroutine levels and surfaces cleanly—with all variables in place—at the first statement. Just to show it can be done, we'll write a dreadful program that does just that. Please don't write programs this way: It's here just to illustrate a point. Remember to type NEW before entering this program.

```
1ØØ IF N>Ø GOTO 13Ø
11Ø PRINT "NAME LIST"
12Ø DIM N$(5Ø)
13Ø PRINT
14Ø PRINT "DO YOU WANT TO --"
15Ø PRINT " 1. ENTER NAMES"
16Ø PRINT " 2. LIST NAMES"
17Ø PRINT " 3. QUIT"
18Ø INPUT "YOUR CHOICE";C
19Ø ON C GOSUB 21Ø,31Ø,35Ø
2ØØ GOTO 13Ø
21Ø PRINT "ENTER EACH NAME"
22Ø PRINT "FOLLOWED BY AN '*'
   CHARACTER"
23Ø PRINT "TO END ENTRY"
24Ø GOSU8 26Ø
25Ø GOTO 24Ø
26Ø INPUT N$
27Ø IF N$="*" OR N=5Ø THEN LOA
   D "DEMO",8
28Ø N=N+1
29Ø N$(N)=N$
3ØØ RETURN
31Ø FOR J=1 TO N
32Ø PRINT N$(J)
33Ø NEXT J
34Ø RETURN
35Ø END
```

Check the program and save it with the filename DEMO; be sure to use that filename, since the program uses it to load itself.

DEMO is a program turned bad, and you should try *not* to get yourself into a similar problem. By the time this program reaches line

210, it's in a subroutine; at line 260, it's nested within a second subroutine. When line 270 discovers that the user wants to exit, the poor programmer doesn't know how to get out. GOTO 130 would be a very bad solution: Jumping out of the routine with GOTO instead of RETURN leaves unprocessed subroutine information on the computer's stack (which can eventually cause an OUT OF MEMORY error). What to do?

The second-best solution (shown here) is to clean up the program with a chain to itself. The best solution is not to get yourself into this kind of mess in the first place.

Chaining can be a useful and powerful technique. There are some rules to remember—especially that of making sure the first program is the biggest—but in general it works quite well. Don't confuse chaining with loading, where one program loads and starts another. In this case, there's no passing of variables; the new program starts clean. We'll talk about loading in next month's installment. ©

CONVERSE WITH YOUR COMPUTER

AT LAST! A FULL IMPLEMENTATION of the original ELIZA program is now available to run on your personal computer!

Created at MIT in 1966, **ELIZA has become** the world's most celebrated artificial intelligence demonstration program. ELIZA is a non-directive psychotherapist who analyzes each statement as you type it in and then responds with her own comment or question—and her remarks are often amazingly appropriate!

Designed to run on a large mainframe, ELIZA has never before been available to personal computer users except in greatly stripped down versions lacking the sophistication which made the original program so fascinating.

Now, our new personal computer version possessing the **FULL** power and range of expression of the original is being offered at the introductory price of only $45. And to let you find out how she does it (or teach her to do more) we have included the complete **SOURCE PROGRAM** (written in BASIC) at no extra cost.

Order your copy of ELIZA today and you'll never again wonder how to respond when you hear someone say, "Okay, let's see what this computer of yours can actually do!"

READ WHAT THE EXPERTS SAY ABOUT OUR VERSION OF ELIZA:

"Much more than a mere game... You'll be impressed with ELIZA... A convincing demonstration of Artificial Intelligence."
—*PC MAGAZINE*

"Delightful entertainment... An ideal medium for showing off your system."
—*MICROCOMPUTING MAGAZINE*

"ELIZA is an astounding piece of software... A fascinating program to use and study."
—*BARON'S MICROCOMPUTER REPORTS*

"ELIZA is a great way to introduce your friends to computers... A very funny party game."
—*PETER A. McWILLIAMS*

"ELIZA is an exceptional program, one that's fun to use, shows off your machine, and has great historical interest."
—*POPULAR COMPUTING MAGAZINE*

"This version of ELIZA is the best we have seen. As a party game, it is unmatched."
—*HOME APPLICATIONS FOR THE C-64*

ELIZA IS AVAILABLE IN THE FOLLOWING FORMATS:

- IBM PC, PCjr, PC-XT and all compatibles
- All Apple II computers (II, II Plus, IIe, IIc)
- Apple Macintosh (Microsoft BASIC required)
- Commodore 64 (specify disk or cassette)
- 5¼ inch or 8 inch disk for all CP/*M* systems

All versions are $45 and include a six page users manual.
Please add $2.00 shipping and handling to all orders
(California residents please add 6½% sales tax)

ARTIFICIAL INTELLIGENCE RESEARCH GROUP
921 North La Jolla Avenue, Dept. M
Los Angeles, CA 90046
(213) 656-7368 (213) 654-2214
MC, VISA and checks accepted

Adding System Power To ST BASIC

Part 1

Kevin Mykytyn, Editorial Programmer

Atari ST BASIC lacks commands for certain operations such as reading the mouse pointer, but it's possible to fill in these gaps by calling system routines with the VDISYS command. In Part 1 of this series, we'll examine the basics of calling VDI routines from BASIC and demonstrate a useful graphics routine. Part 2 will show how to read the mouse pointer with VDISYS and present a program for creating your own custom mouse pointers.

If you own an Atari ST, you've probably heard at least two of the three-letter acronyms associated with the computer: TOS stands for Tramiel Operating System—a huge system program which, at the most fundamental level, allows the computer to function. And GEM stands for Graphics Environment Manager, a separate system program that handles the ST's graphics-oriented desktop. GEM, in turn, consists of three separate parts: the VDI (Virtual Device Interface), a low-level graphics interface that also handles mouse input; the AES (Application Environment Services), which uses the VDI to manage data and the desktop; and GEMDOS, which handles disk operations.

Interesting, you may say, but what's the point? For most BASIC programming, you needn't worry about TOS, GEM, VDI, AES, or GEMDOS, any more than the average driver needs to know exactly how an auto engine works. These system programs are the invisible machinery that makes everything else happen.

However, as you may have discovered, ST BASIC lacks commands to do certain tasks, such as drawing a circle or sensing the position of the mouse pointer. That's what makes one of these strange-sounding programs—the VDI—an invaluable asset for the BASIC programmer. The VDI holds a treasure trove of system routines which can do everything from drawing boxes and circles to rotating character fonts and manipulating raster blocks. With ST BASIC's VDISYS command, you can access all of these routines—which compensates in large part for the missing ST BASIC commands.

VDISYS To The Rescue

In simplest terms, the VDISYS command calls (activates) a VDI system routine to do a task that would be difficult or impossible to perform in BASIC. Furthermore, these system routines execute very quickly—a real plus when you're working with graphics. Whether executed in immediate or program mode, the VDISYS command always takes this general form:

```
VDISYS(x)
```

In this example a simple variable named *x* appears in the parentheses. It doesn't matter what value this variable represents; it's a *dummy* parameter, needed only to satisfy the syntax of the command. Don't try to enter this command yet—if you do, there's a good chance you'll see the mushroom cloud symbol that signals a system crash. A certain amount of preparation is always needed before you execute VDISYS.

When a VDISYS command is executed, control passes from your BASIC program to an internal VDI handler, which eventually passes control to the VDI routine itself. But first the VDI handler looks at certain sections of the computer's memory, called *parameter blocks.* The data in the parameter blocks tells the handler which particular VDI routine you want to execute. There's also other information that the VDI routine itself will need. If you don't supply all the information needed to call a routine, the VDI handler can't carry out your request.

VDI Opcodes

The first thing you must tell the computer is which VDI routine you want to call. Each VDI routine is identified by a unique *opcode number.* For instance, the VDI routine used in the program below has the opcode 11. This is a generalized shape-drawing routine. (There are hundreds of VDI opcodes and associated parameters, so we don't have room in this article for a listing. But you can find a 42-page list of selected VDI opcodes in COMPUTE!'s *ST Programmer's Guide,* available from COMPUTE! Publications.)

Once you know a VDI routine's opcode number, that value must be POKEd into a special place in memory defined by the reserved variable CONTRL. Try typing PRINT CONTRL in immediate mode; even if you haven't given this variable any value, the computer prints a number on the screen. ST BASIC always predefines CONTRL along with several similar variables. The CONTRL variable represents an actual location in memory.

Since the system automatically substitutes this location for the keyword CONTRL, you don't have to memorize a series of numbers or worry about where this parameter block really resides. To select VDI routine 11, for instance, you simply execute POKE CONTRL,11.

How Many Corners?

Once you've POKEd the VDI opcode 11 into CONTRL, you must tell the computer how many vertices (corners) are needed to define the graphic shape you want to draw. Regular geometric shapes require different numbers of vertices. A triangle, for instance, requires a minimum of three corners. A rectangle, on the other hand, can be defined with only two—the upper left corner and the lower right one. Of course, a rectangle has a total of four corners, but the total is not what we're looking for. The computer cares only about the *minimum* number of vertices it takes to draw the shape in question. After you determine how many vertices are needed, that value is POKEd into the location defined by CONTRL +2. For example, in line 30 of the program below, the statement POKE CONTRL+2,2 tells the computer that you want to draw a rectangle (defined by only two corners).

Notice that the second POKE is directed two bytes higher in memory than the first. Now you can see the parameter block begin to take shape: It's simply a segment of memory where you place a collection of values. The first byte of the parameter block is defined by CONTRL, and the remaining locations are defined as even-numbered offsets above that starting spot (CONTRL+2, CONTRL+4, and so forth).

The particular routine used in this program (termed a *generalized drawing primitive*) contains several subroutines (also called *subfunctions*), each of which performs a different drawing task. To choose a subroutine, you must POKE its identifying number (called the *primitive ID*) into the location defined by CONTRL+10. In this case we want to use the bar-drawing subroutine, whose primitive ID happens to be 1. So in line 40 of the program, we POKE CONTRL+10,1.

PTSIN And INTIN

The next step is to tell the VDI handler where to place the graphic shape. Recall that you told the computer earlier how many vertices it takes to define the shape. To position the shape on the screen, you must now tell VDI where to put each vertex. This is done by POKEing horizontal (X) and vertical (Y) coordinate values into a second parameter block area.

The second parameter block begins at a memory location defined by the reserved variable PTSIN (*Points Input*). Again, you don't need to know the actual memory locations involved, since the computer keeps track of them for you. All you need to do is POKE the correct numbers into PTSIN (and even-numbered adjacent locations, in some cases).

Lines 50–80 of the example program perform this job by POKEing the bar's X and Y coordinates into memory. The X coordinate of the first point is POKEd into PTSIN; the first point's Y coordinate goes into PTSIN+2; the X coordinate of the second point goes into PTSIN+4, and so on. Keep in mind that you must supply a *pair* of coordinate values for every point that you defined in CONTRL+2.

A third parameter block, beginning at the address defined by the reserved variable INTIN, is used to pass *attribute values,* if any are required by the current subroutine. The term attribute is a catch-all that can include many different parameters—colors, rotation values, a style index, or whatever—depending on which subroutine is called. Since the subroutine used in this program requires no attributes, we don't need to POKE any values in this segment of memory. As a signal to the VDI handler that no attributes are involved, we must also POKE a zero into location CONTRL+6; this location tells the system how many attribute values to read from the INTIN parameter block.

After all of the required values have been POKEd into memory, line 90 of the example program executes the VDISYS command, which calls the VDI routine and draws a bar on the screen. This may seem like an enormous amount of preparation for such a simple task (which some other computers can do with a single BASIC statement). On the other hand, it's better than not being able to draw a bar at all. You can cut down on the bulkiness of the code by writing setup subroutines that contain all the necessary overhead.

Bar Drawing

```
10 fullw 2:clearw 2:color 2,2,2
20 poke contrl,11 'VDI opcode
30 poke contrl+2,2 'number of vertices
35 poke contrl+6,0 'number of attributes
40 poke contrl+10,1 'primitive ID of bar
   command
50 poke ptsin,50 'x coordinate of top left
   corner
60 poke ptsin+2,50 'y coordinate of top
   left corner
70 poke ptsin+4,100 'x coordinate of
   bottom right corner
80 poke ptsin+6,100 'y coordinate of
   bottom right corner
90 vdisys (0)
```

General Drawing Routine

Though every VDI call requires several preparatory steps, each individual step is easy to perform. As should be apparent by now, there's nothing mystical about the process—all you need to do is leave the right pieces of information in places where the computer can find them, then signal that you want the job done. The real work is done by the system itself.

Though the general procedure is the same in every case, each VDI routine requires different types and amounts of information. One of the most useful VDI routines is the generalized drawing primitive used in the example program. Table 1 summarizes the POKEs you need to call this routine.

Table 1: Generalized Drawing Primitive

```
POKE CONTRL, 11
POKE CONTRL+2, number of vertices
POKE CONTRL+6, number of attributes
POKE CONTRL+10, subfunction number
                (primitive ID)
```

Again, CONTRL receives the opcode number of the VDI routine; CONTRL+2 the number of vertices in the desired shape; CONTRL+6 the number of attributes (if any); and CONTRL+10 the primitive ID for the subroutine you want. This particular VDI routine is extremely versatile and can draw pie-shaped segments, ellipses, filled or empty rounded rectangles, and other graphic images, including text. Table 2 lists the primitive IDs for each of this routine's subroutines.

Table 2: Drawing Subroutines

Primitive ID	Subroutine
1	bar
2	circle
3	arc
4	pie
5	ellipse
6	elliptical arc
7	elliptical pie
8	rounded rectangle
9	filled rounded rectangle
10	justified graphics text

To select a specific subroutine, find its primitive ID in the leftmost column of Table 2, then POKE that value into location CONTRL+10. Table 3 summarizes the POKEs needed to set up the second and third parameter blocks (PTSIN and INTIN). Remember, the value POKEd into CONTRL+2 (number of vertices) determines how many X-Y coordinate pairs you must POKE into the PTSIN parameter block. The X and Y coordinates for the first vertex go into PTSIN and PTSIN+2; the second X-Y coordinate pair goes into PTSIN+4 and PTSIN+6, and so forth.

Table 3: PTSIN And INTIN Parameter Blocks

```
POKE PTSIN, X coordinate of first vertex
              (rectangle)
            X coordinate of center
              (circle, ellipse)
POKE PTSIN+2, Y coordinate of first
                vertex (rectangle)
              Y coordinate of center
                (circle, ellipse)
POKE PTSIN+4, X coordinate of second
                vertex (rectangle)
              X radius for ellipse
POKE PTSIN+6, Y coordinate of second
                vertex (rectangle)
POKE PTSIN+8, radius (circle only)
POKE PTSIN+12, radius (circular arc or
                pie only)

POKE INTIN, start angle for arcs and pies
POKE INTIN+2, end angle for arcs and
              pies
```

To draw a circle, ellipse, arc, or pie-shape segment, POKE X and Y coordinates for the shape's center point into PTSIN and PTSIN+2. A simple circle requires a radius value in PTSIN+8; arcs and pie shapes built from a part of a circle require a radius value in PTSIN+12. To draw an ellipse, or an arc or pie shape built from part of an ellipse, POKE the shape's X radius in PTSIN+4 and its Y radius into PTSIN+6.

Most of these subfunctions don't require any attribute values. To draw arcs or pie shapes, however, you must POKE two attribute values into INTIN and INTIN+2 to define starting and ending angles, respectively. Since the angle values are specified in tenths of a degree, not in whole degrees, these parameters can range from 0–3600. The starting angle specifies where you want the rounded portion of the arc or pie segment to begin, and the ending angle shows where that portion should stop. The statement POKE CONTRL+6,2 signals that you're passing two attribute values to the VDI.

As you'll learn from experimenting with these routines, VDISYS opens the gateway to a wide variety of graphics capabilities. Once you become familiar with the setup process, you'll probably find yourself using VDISYS more and more. In part 2 of this article, we'll look at VDISYS in more detail, and present a program that lets you create a custom shape for your ST's mouse pointer. ©

Mousify Your Applesoft Programs
Part 2

Lee Swoboda

*Part 1 of this series (*COMPUTE!, *March 1985) provided an Applesoft program allowing an AppleMouse, joystick, or game paddles to point to text on the screen. This month, Part 2 demonstrates more advanced mouse operations such as defining a text area and deleting, copying, or restoring the defined text. The example programs run with either DOS 3.3 or ProDOS. Although a mouse works best, you can substitute a joystick or game paddles.*

Mouse-controlled programs must perform a number of functions in addition to simple pointing. The programs following this article provide several of these important capabilities:

- *Define Text.* Use the mouse to highlight a block of text, which can then be copied or deleted (typical word processing operations).
- *Copy Text.* Copy highlighted text to a buffer without deleting it from the screen.
- *Delete Text.* Delete highlighted text and save it in a buffer.
- *Insert Text.* Restore previously copied or deleted text at a new point on the screen.
- *Cancel.* Undo highlighting if you wish to abort a copy or delete operation.
- *Delete a Character.* Delete the character under the cursor.
- *Delete to End of Line.* Delete text from the cursor to the end of the line.
- *Find Mouse.* Locate the mouse interface.

Getting Started

Enter and save Program 1, which is an expanded and modified version of the program published in Part 1. It works in either DOS 3.3 or ProDOS; if you're using ProDOS, change line 115 as shown here:

```
115 HIMEM: 36352
```

Program 2 creates a binary file named MOUSEY which contains machine language routines used by Program 1. The MOUSEY file created by Program 2 must be present on disk whenever you run Program 1. (It's not necessary to have Program 2 itself on the disk with Program 1, just a copy of the binary file created by Program 2.) Be sure to save a copy of Program 2 so you'll be able to create new copies of MOUSEY whenever needed.

Program 3 creates a short text file which we'll use in the following example. If you're using a joystick instead of a mouse, refer to the additional instructions under "Joystick Modifications" below. When you are ready to proceed, your disk should contain a copy of Program 1, a file named MOUSEY (created by Program 2), and a file named TEXT (created by Program 3).

When you run Program 1, the screen looks like this:

```
          ENTER INFORMATION

FIRST NAME  ....  COMPUTE!
LAST NAME  .....  READER SERVICE
ADDRESS ........  P.O. BOX 50950
CITY ...........  DES MOINES
STATE ..........  IA 50950
ZIP ............
TELEPHONE  .....  1-800-346-6767

  COPY  DELETE  INSERT  CANCEL
     ERASE  QUIT  DONE  HELP
```

This screen simulates what you might see in a simple address book program. We have introduced an intentional error by putting the zip code entry on the same line as the state entry. Let's correct the error for a quick demonstration of a few mouse features. Move the mouse cursor to the first number in the zip code, then press and *hold* the mouse button down while moving the mouse to the right. The computer highlights the zip code in inverse video. Keep moving the mouse until all the numbers in the zip code are highlighted, then release the mouse button.

At this point, the highlighted text area has been defined. Now move the mouse pointer to the word DELETE in the strip menu at the bottom of the screen and press the mouse button. The computer erases the highlighted zip code from the screen. Don't worry—the information hasn't been lost. Whenever you delete text, the program stores it in a temporary memory buffer.

Now let's put the zip code data back where it belongs. Move the mouse pointer to the beginning of the next screen line (directly under the I in IA), then press the mouse button. The computer moves the cursor to that line. Next, move the mouse pointer to the word INSERT and press the button again. The zip code data reappears in the desired screen area.

Mouse Editing Functions

Here is a more detailed description of the mouse-editing functions demonstrated in Program 1:

Mouse pointer and text cursor. The rapidly blinking caret symbol (^) is the mouse pointer, which you can move around the text screen with the mouse. When the pointer passes over a character, the character blinks rapidly. The flashing rectangle shows the position of the text cursor. When the cursor passes over a character, the character changes temporarily to flashing uppercase. There are three different ways to move the text cursor:

- Move the mouse pointer to the spot where you want the text cursor to go, then press the mouse button.
- Use the arrow keys as you would in Applesoft BASIC (the Apple II uses CTRL-J and CTRL-K to move up and down, respectively).
- Press RETURN to move the cursor to the beginning of the next screen line. If the cursor is already on the bottom line, it moves to the top. Pressing RETURN does not erase the text to the right of the cursor.

Enter text. Text is entered as usual, by pressing any letter, number, or punctuation key. Lowercase letters are automatically converted to uppercase.

Define text. Before text can be copied or deleted, you must define it. Move the mouse pointer to the upper-left corner of the text you want to define, then press and hold the mouse button. While pressing the button down, drag the mouse pointer to the lower-right corner of the desired area. The computer marks the defined area by highlighting every character with inverse video. Now release the button: The area is defined, and you may proceed to the Cancel, Delete, or Copy options.

Delete text. To delete a text area that you previously defined, move the mouse pointer to DELETE in the strip menu at the bottom of the text screen, then press the button. The computer blanks out the highlighted portion of the screen and stores the first 200 characters of the defined area in a temporary buffer for later use.

Copy text. To copy a text area that you have previously defined, move the pointer to COPY in the strip menu, then press the button. The computer stores the first 200 characters of the defined area in a temporary buffer. Unlike the Delete operation, Copy does not blank out the defined area.

Insert text. To insert text that you previously copied or deleted, move the pointer to the spot where you want to insert text, then press the button to locate the cursor at that spot. Now move the pointer to INSERT in the strip menu and press the button again. The computer inserts the text, using the text cursor position as a starting point. Note that the inserted text overwrites whatever else was in the affected area. You can insert only the most recently copied or deleted text.

Cancel. If you define a block of text and then decide not to copy or delete it, move the pointer to CANCEL in the strip menu and press the button. The highlighting disappears, and the text is no longer defined.

Editing keys. Press CTRL-D (or DELETE on the Apple IIc and IIe) to delete the character under the cursor. The remaining characters in that line move one space to the left. You can also press CTRL-X to delete every character from the present cursor position to the end of the line.

Try out the various editing functions. When you've tried everything, move the mouse pointer to DONE in the strip menu and press the button. The demonstration program ends with a routine that reads the current data directly from the video screen.

Since the Copy, Delete, Insert, and Cancel commands are written in BASIC, they may take a second or two to complete if you define a large text area. Though BASIC can't perform such operations very fast, these routines are far easier for you to customize than if they had been written in machine language. If the slowness bothers you, just imagine how long it would take to delete the same amount of text with your trusty pink eraser.

Joystick Modifications

If you don't own a mouse, you can substitute a joystick. Delete lines 120, 130, 10001–10090, 10200, 20220, and 44000–44050 from Program 1; then add or modify the lines in Program 4. The joystick moves the mouse pointer around the screen, and the joystick button substitutes for the mouse button.

Since the joystick was designed for a different purpose, its performance doesn't equal that of a mouse. But it costs a lot less.

How The Program Works

The machine language routine contained in the MOUSEY file simply highlights text by changing every character between the text cursor and mouse pointer to inverse video. All the other functions are carried out by the BASIC routines in Program 1.

After you define a block of text, lines 35000–44050 act on the highlighted area. The Copy routine (36000–36180) converts each character in the defined area to normal video and stores it in a temporary text buffer. This buffer lies in locations 775–1000 ($307–$3E8), a normally unused region.

The Delete routine (37000–37180) is similar to Copy and uses the same temporary buffer, but replaces each character in the defined area with a blank space.

The Insert routine (38000–38100) moves text from the temporary buffer back to the video screen, beginning at the current location of the text cursor.

Lines 39000–40000 contain the Cancel routine, which aborts copy or delete operations. You can also cancel a definition by pressing any key.

The routine at lines 41000–41070 deletes a single character; lines 42000–42060 erase all or part of the current line.

Here are some other useful entry points in the program (note that each of these routines ends with a GOTO rather than GOSUB):

Line	Purpose
10120	read mouse
10300	position mouse pointer
10420	keyboard input
10570	position cursor

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing in Programs" in this issue of COMPUTE!.

Program 1: Advanced Mousification

```
81 115 HIMEM: 37375
DF 12Ø GDSUB 44Ø1Ø
D7 13Ø MI = 2Ø: REM MDUSE SENSIT
       IVITY
5A 14Ø D$ = CHR$ (4)
EB 145 PRINT D$"BLOAD MOUSEY"
8C 15Ø REM
87 16Ø REM READ DATA FILE
90 17Ø REM
CB 1BØ PRINT D$"DPEN TEXT"
```

```
32 190 PRINT D$"READ TEXT"
6D 200 INPUT NF$,NL$,AD$,CI$,ST$
       ,ZI$,TE$
CD 210 PRINT D$"CLOSE TEXT"
87 220 REM
25 230 REM DATA ENTRY SCREEN
88 240 REM
4F 250 HOME
4D 260 Y1 = 4:X1 = 15:CØ = 16Ø
35 270 INVERSE
D7 280 PRINT "             ENTER
       INFORMATION            "
24 285 VTAB 23: PRINT "       COP
       Y  DELETE  INSERT  CANCEL
          "
9B 290 VTAB 24: PRINT "          E
       RASE  QUIT  DONE  HELP
          ";
C6 300 NORMAL
31 310 VTAB 4: HTAB 1
F4 320 PRINT "FIRST NAME .."
C6 330 PRINT "LAST NAME ..."
3C 340 PRINT "STREET ......"
D6 350 PRINT "CITY ........"
1F 360 PRINT "STATE ......."
36 370 PRINT "ZIP ........."
17 380 PRINT "TELEPHONE ..."
3A 390 VTAB 19: HTAB 1Ø: INVERSE
        : PRINT "^";: NORMAL
81 400 PRINT " IS MOUSE POINTER"
3C 410 VTAB 21: HTAB 14: INVERSE
        : PRINT " ";: NORMAL
38 420 PRINT " IS CURSOR"
26 430 VTAB 4
5E 440 HTAB 15: PRINT NF$
66 450 HTAB 15: PRINT NL$
D9 460 HTAB 15: PRINT AD$
E1 470 HTAB 15: PRINT CI$
F6 480 HTAB 15: PRINT ST$
71 490 HTAB 15: PRINT ZI$
59 500 HTAB 15: PRINT TE$
73 9999 REM #1ØØØØ
19 1ØØØØ REM
E6 1ØØØ1 REM --------------
29 1ØØ1Ø REM MOUSE ROUTINES
E6 1ØØ2Ø REM --------------
37 1ØØ4Ø REM
A4 1ØØ5Ø REM TURN MOUSE "ON"
49 1ØØ6Ø REM
A8 1ØØ7Ø PRINT D$"PR#"5Ø: PRINT
       CHR$ (1)
C8 1ØØ8Ø PRINT D$"PR#Ø"
89 1ØØ9Ø PRINT D$"IN#"5Ø
17 1Ø1ØØ GOTO 1Ø59Ø
25 1Ø11Ø REM
65 1Ø12Ø REM DETERMINE POSITION
91 1Ø13Ø REM OF MOUSE
3D 1Ø14Ø REM
1C 1Ø15Ø VTAB 1: HTAB 4Ø
77 1Ø16Ø INPUT "";XØ,YØ,BØ
7D 1Ø17Ø IF BØ < Ø THEN 1Ø44Ø: R
       EM KEY PRESSED?
D8 1Ø18Ø YØ = INT (YØ / MI) + 1
78 1Ø19Ø IF YØ > 24 THEN YØ = 24
64 1Ø2ØØ XØ = INT (XØ / MI) + 1
75 1Ø21Ø IF XØ > 4Ø THEN XØ = 4Ø
6A 1Ø215 IF BØ = 2 AND XØ > 2Ø A
       ND XØ < 27 AND YØ = 23
       THEN 38Ø1Ø
6D 1Ø216 IF BØ = 2 AND SW = Ø TH
       EN X2 = XØ:Y2 = YØ: POK
       E 768,Y2: POKE 769,X2:X
       3 = XØ:Y3 = YØ
98 1Ø217 IF BØ = 3 AND SW = 2 TH
       EN SW = 3
31 1Ø218 ON SW GOTO 35Ø1Ø,35Ø2Ø,
       1Ø315
8B 1Ø22Ø IF BØ > 1 THEN 1Ø32Ø: R
       EM BUTTON PRESSED?
D9 1Ø225 IF X2 < > XØ OR Y2 < >
       YØ THEN SW = 1: GOTO 35
       Ø1Ø
89 1Ø23Ø IF YØ = 24 THEN 2ØØ3Ø
```

```
63 1Ø24Ø Y1 = YØ:X1 = XØ
78 1Ø25Ø POKE VØ,CØ
E8 1Ø255 IF CØ < 128 THEN POKE V
       Ø,CØ + 128
48 1Ø26Ø CØ = C2
F2 1Ø27Ø GOSUB 1Ø8ØØ
F2 1Ø28Ø GOTO 1Ø62Ø
69 1Ø29Ø REM
ED 1Ø3ØØ REM POSITION MOUSE POIN
       TER
2D 1Ø31Ø REM
DD 1Ø315 IF BØ = 2 AND YØ = 23 T
       HEN 2ØØ81
86 1Ø32Ø IF VØ = V1 THEN C2 = C1
88 1Ø33Ø POKE V1,C2
A2 1Ø34Ø V1 = 1Ø23 + 128 * (YØ -
        1) + XØ
3F 1Ø35Ø IF YØ > 8 THEN V1 = V1
       - 984
9C 1Ø36Ø IF YØ > 16 THEN V1 = V1
        - 984
27 1Ø37Ø C2 = PEEK (V1)
64 1Ø38Ø POKE V1,16Ø
BA 1Ø39Ø IF C2 = 16Ø THEN POKE V
       1,3Ø
C2 1Ø4ØØ GOTO 1Ø15Ø
31 1Ø41Ø REM
81 1Ø42Ø REM KEYBOARD INPUT
41 1Ø43Ø REM
F9 1Ø44Ø C3 = PEEK ( - 16384)
77 1Ø45Ø POKE - 16368,Ø
DC 1Ø455 IF C3 > 223 THEN C3 = C
       3 - 32: REM CONVERT TO
       UPPER CASE
7C 1Ø456 IF SW > Ø THEN GOSUB 39
       Ø1Ø
4B 1Ø46Ø IF C3 > 159 THEN 1Ø71Ø
63 1Ø465 IF C3 = 132 OR C3 = 225
        THEN IF X1 > 14 AND Y1
        > 3 AND Y1 < 11 THEN G
       OSUB 41Ø1Ø
BA 1Ø466 IF C3 = 152 THEN IF X1
       > 14 AND Y1 > 3 AND Y1
       < 11 THEN GOSUB 42Ø1Ø
CD 1Ø47Ø IF C3 = 141 THEN X1 = 1
       5:Y1 = Y1 + 1: IF Y1 >
       1Ø THEN Y1 = 4: REM RET
       URN KEY
59 1Ø48Ø IF C3 = 138 THEN Y1 = Y
       1 + 1: REM DOWN ARROW
C2 1Ø49Ø IF C3 = 139 THEN Y1 = Y
       1 -1: REM UP ARROW
8F 1Ø5ØØ IF C3 = 149 THEN X1 = X
       1 + 1: REM RIGHT ARROW
71 1Ø51Ø IF C3 = 136 THEN X1 = X
       1 - 1: REM LEFT ARROW
56 1Ø52Ø IF Y1 > 24 THEN Y1 = 24
DC 1Ø53Ø IF Y1 < 1 THEN Y1 = 1
9C 1Ø54Ø IF X1 > 4Ø THEN X1 = 4Ø
E8 1Ø55Ø IF X1 < 1 THEN X1 = 1
5D 1Ø56Ø REM
86 1Ø57Ø REM POSITION CURSOR
6D 1Ø58Ø REM
A4 1Ø59Ø POKE VØ,CØ
CA 1Ø6ØØ GOSUB 1Ø8ØØ
42 1Ø61Ø CØ = PEEK (VØ)
9E 1Ø62Ø IF VØ = V1 THEN CØ = C2
44 1Ø63Ø REM CHANGE TO FLASHING
       CHARACTER
87 1Ø64Ø C1 = CØ
23 1Ø65Ø IF C1 > 127 THEN C1 = C
       1 - 64
7F 1Ø66Ø IF C1 > 64 THEN C1 = C1
        - 64
D9 1Ø67Ø IF C1 > 95 THEN C1 = C1
        - 32
48 1Ø68Ø IF C1 < 64 THEN C1 = C1
        + 64
C8 1Ø69Ø POKE VØ,C1
CE 1Ø7ØØ GOTO 1Ø15Ø
68 1Ø71Ø IF X1 < 15 OR Y1 < 4 OR
        Y1 > 1Ø THEN 1Ø15Ø
DE 1Ø72Ø GOSUB 1Ø8ØØ
```

```
DC 1Ø73Ø POKE VØ,C3
51 1Ø74Ø CØ = C3
CE 1Ø75Ø IF VØ = V1 THEN C2 = C3
BC 1Ø76Ø X1 = X1 + 1: IF X1 > 39
        THEN X1 = 39
67 1Ø77Ø GOTO 1Ø59Ø
16 1Ø78Ø REM CALCULATE VØ
6E 1Ø79Ø REM (VIDEO BUFFER ADDRE
       SS)
6B 1Ø8ØØ VØ = 1Ø23 + 128 * (Y1 -
        1) + X1
2B 1Ø81Ø IF Y1 > 8 THEN VØ = VØ
       - 984
7F 1Ø82Ø IF Y1 > 16 THEN VØ = VØ
        - 984
8B 1Ø83Ø RETURN
9A 19999 REM #2ØØØØ
1A 2ØØØØ REM
AE 2ØØ1Ø REM STRIP MENU
2A 2ØØ2Ø REM
C2 2ØØ3Ø PRINT D$"IN#Ø"
CB 2ØØ4Ø IF XØ > 8 AND XØ < 14 T
       HEN NF$ = "":NL$ = "":A
       D$ = "":CI$ = "":ST$ =
       "":ZI$ = "":TE$ = "": G
       OTO 25Ø
1F 2ØØ5Ø IF XØ > 15 AND XØ < 2Ø
       THEN HOME : END
73 2ØØ6Ø IF XØ > 21 AND XØ < 26
       THEN 3ØØ3Ø
7A 2ØØ7Ø IF XØ > 27 AND XØ < 32
       THEN 2Ø1ØØ
71 2ØØ8Ø VTAB 1: HTAB 4Ø: PRINT
       D$"IN#"5Ø: GOTO 1Ø15Ø
5C 2ØØ81 IF XØ > 6 AND XØ < 11 T
       HEN GOSUB 36Ø1Ø: GOTO 1
       Ø59Ø
B2 2ØØ82 IF XØ > 12 AND XØ < 19
       THEN GOSUB 37Ø1Ø: GOTO
       1Ø59Ø
5F 2ØØ83 IF XØ > 28 AND XØ < 35
       THEN GOSUB 39Ø1Ø: GOTO
       1Ø59Ø
34 2ØØ84 GOTO 1Ø15Ø
17 2ØØ9Ø REM HELP TEXT
CD 2Ø1ØØ VTAB 12: HTAB 1
8A 2Ø11Ø PRINT "THE FLASHING REF
       LEX (^) IS THE MOUSE"
75 2Ø12Ø PRINT "POINTER AND THE
       FLASHING RECTANGLE IS"
48 2Ø13Ø PRINT "THE CURSOR.  TO
       MOVE THE CURSOR TO THE"
36 2Ø14Ø PRINT "ENTRY YOU WANT T
       O CHANGE, USE THE ARROW
       "
4E 2Ø15Ø PRINT "KEYS OR USE THE
       MOUSE TO MOVE THE MOUSE
       "
47 2Ø16Ø PRINT "POINTER, THEN PR
       ESS THE MOUSE BUTTON TO
       "
E6 2Ø17Ø PRINT "MOVE THE CURSOR
       TO THAT POINT.  TYPE"
EA 2Ø18Ø PRINT "NEW OR CORRECTED
        DATA, THEN MOVE THE"
31 2Ø19Ø PRINT "MOUSE CURSOR TO
       'DONE' IN THE MENU"
4A 2Ø2ØØ PRINT "BELOW AND PRESS
       THE MOUSE BUTTON TO"
D4 2Ø21Ø PRINT "ACCEPT THE ENTRI
       ES ABOVE."
D9 2Ø22Ø PRINT D$"IN#"5Ø
D3 2Ø23Ø GOTO 1Ø15Ø
9D 29999 REM #3ØØØØ
1B 3ØØØØ REM
28 3ØØ1Ø REM EXAMPLE
28 3ØØ2Ø REM
A1 3ØØ3Ø Y1 = 4: GOSUB 63Ø5Ø:NF$
        = A$
2C 3ØØ4Ø Y1 = 5: GOSUB 63Ø5Ø:NL$
        = A$
91 3ØØ5Ø Y1 = 6: GOSUB 63Ø5Ø:AD$
        = A$
```

```
1C 30060 Y1 = 7: GOSUB 63050:CI$
        = A$
E9 30070 Y1 = B: GOSUB 63050:ST$
        = A$
11 30080 Y1 = 9: GOSUB 63050:ZI$
        = A$
17 30090 Y1 = 10: GDSUB 63050:TE
       $ = A$
2E 30100 REM GO TD REMAINOER DF
       YDUR PROGRAM
1C 30110 REM FDR EXAMPLE ...
36 30120 HDME
5E 30130 VTAB 10
EE 30140 PRINT NF$" "NL$
38 30150 PRINT AD$
BD 30160 PRINT CI$", "ST$" "ZI$
9C 30170 PRINT TE$
CA 30180 CALL - 19B: CALL - 19B
89 30190 END : REM END OF EXAMPL
       E
EC 35000 REM HIGHLIGHT TEXT
AD 35010 POKE V0,C1 - 64:SW = 2
AC 35020 IF 80 > 1 THEN 10150
4D 35030 IF X3 < X0 THEN X3 = X0
D5 35040 IF Y3 < Y0 THEN Y3 = Y0
88 35050 PDKE 770,Y3: POKE 771,X
       3
8D 35060 POKE 772,Y0: PDKE 773,X
       0
FC 35070 CALL 37376
28 35080 Y3 = Y0:X3 = X0
D7 35090 BOTD 10150
9C 36000 REM CDPY
6D 36010 P3 = 775
3B 36020 FOR II = Y2 TO Y3
8A 36030 FOR JJ = X2 TO X3
AC 36040 GOSUB 40020
39 36050 C3 = PEEK (V2) + 128
96 36060 IF C3 < 160 THEN C3 = C
       3 + 64
37 36070 IF C3 > 223 THEN C3 = C
       3 - 64
FE 36080 POKE V2,C3
CD 36090 IF P3 > 1000 THEN 36120
BA 36100 POKE P3,C3
DA 36110 P3 = P3 + 1
D5 36120 NEXT JJ
3E 36130 POKE P3,141
EE 36140 IF P3 < 1001 THEN P3 =
       P3 + 1
E7 36150 NEXT II
87 36160 POKE P3,255
BB 36170 SW = 0
A5 361B0 RETURN
E8 37000 REM DELETE
62 37010 P3 = 775
3A 37020 FOR II = Y2 TO Y3
BC 37030 FOR JJ = X2 TO X3
AE 37040 GOSUB 40020
3B 37050 C3 = PEEK (V2) + 128
98 37060 IF C3 < 160 THEN C3 = C
       3 + 64
39 37070 IF C3 > 223 THEN C3 = C
       3 - 64
6C 37080 POKE V2,160
D7 37090 IF P3 > 1000 THEN 37120
BC 37100 POKE P3,C3
DC 37110 P3 = P3 + 1
D7 37120 NEXT JJ
4D 37130 POKE P3,141
F6 37140 IF P3 < 1001 THEN P3 =
       P3 + 1
E9 37150 NEXT II
89 37160 POKE P3,255
D4 37170 SW = 0:C0 = 160
A7 37180 RETURN
D7 38000 REM INSERT
64 38010 P3 = 775
05 38020 II = Y1:JJ = X1
0D 38030 C3 = PEEK (P3)
F1 38040 IF C3 = 141 THEN II = I
        I + 1:JJ = X1:P3 = P3 +
         1: GOTO 38030
72 38050 IF II > 22 OR JJ > 40 T
       HEN 38090
DD 38060 IF C3 = 255 THEN SW = 0
       :C0 = PEEK (V0): GOTO 1
       0590
CB 38070 GDSUB 40020
03 38080 POKE V2,C3
4E 38090 JJ = JJ + 1:P3 = P3 + 1
C4 38100 GDTD 38030
C3 39000 REM CANCEL
36 39010 FDR II = Y2 TD Y3
8B 39020 FDR JJ = X2 TD X3
AA 39030 GDSUB 40020
37 39040 C3 = PEEK (V2) + 128
94 39050 IF C3 < 160 THEN C3 = C
       3 + 64
35 39060 IF C3 > 223 THEN C3 = C
       3 - 64
FC 39070 POKE V2,C3
43 39080 NEXT JJ,II
C2 39090 SW = 0
56 40000 RETURN
24 40010 REM
78 40020 V2 = 1023 + 128 * (II -
        1) + JJ
4B 40030 IF II > 8 THEN V2 = V2
       - 9B4
A6 40040 IF II > 16 THEN V2 = V2
        - 984
7E 40050 RETURN
7F 41000 REM DELETE A CHARACTER
83 41010 GOSUB 43010
96 41020 FOR II = V0 TO V2 - 1
9B 41030 POKE II, PEEK (II + 1)
D2 41040 NEXT II
49 41050 POKE V2,160
57 41060 C0 = PEEK (V0)
98 41070 RETURN
BF 42000 REM DELETE TO ENO OF LI
       NE
85 42010 GOSUB 43010
76 42020 FOR II = V0 TO V2
7D 42030 POKE II,160
04 42040 NEXT II
FA 42050 C0 = 160
8A 42060 RETURN
22 43000 REM
CF 43010 V2 = 1063 + 128 * (Y1
        1)
2E 43020 IF Y1 > 8 THEN V2 = V2
       - 9B4
94 43030 IF Y1 > 16 THEN V2 = V2
        - 984
7C 43040 RETURN
DB 44000 REM FINO MOUSE
56 44010 FOR S0 = 0 TO 6
AA 44020 IF PEEK (49420 + (256 *
        S0)) = 32 AND PEEK (49
       659 + (256 * S0)) = 214
        THEN S0 = S0 + 1: RETU
       RN
8B 44030 NEXT S0
B2 44040 PRINT "I CAN'T FINO A M
       OUSE INTERFACE CARO" CH
       R$ (7) CHR$ (7)
6B 44050 ENO
A5 62999 REM #63000
24 63000 REM
2C 63010 REM SUBROUTINE TO "REAO
       "
1F 63020 REM STRINGS FROM THE
B3 63030 REM VIOEO BUFFER
44 63040 REM
0F 63050 VTAB 24: FLASH : PRINT
       "  WORKING ...
                            ";: NO
       RMAL : VTAB 1: HTAB 1
C9 63060 A$ = ""
FC 63070 REM CALCULATE V0
55 63080 REM (VIOEO BUFFER AODRE
       SS)
A5 63090 V0 = 1037 + 128 * (Y1 -
        1)
12 63100 IF Y1 > 8 THEN V0 = V0
       - 984
66 63110 IF Y1 > 16 THEN V0 = V0
        - 9B4
2F 63120 FDR I = 1 TD 25
67 63130 C0 = PEEK (V0 + I)
AD 63135 IF C0 < 12B THEN C0 = C
       0 + 128
7B 63136 IF C0 < 160 THEN C0 = C
       0 + 64
19 63137 IF C0 > 223 THEN C0 = C
       0 - 64
DD 63140 IF C0 = 160 AND PEEK (V
       0 + I + 1) = 160 THEN 6
       3190: REM ENO IF TWO BL
       ANKS
F9 63160 IF C0 > 12B THEN C0 = C
       0 - 128
F5 63170 A$ = A$ + CHR$ (C0)
D5 631B0 NEXT I
C2 63190 IF RIGHT$ (A$,1) = CHR$
        (32) THEN A$ = LEFT$ (
       A$, LEN (A$) - 1): GOTO
        63190: REM REMDVE TRAI
       LING BLANKS
42 63195 IF A$ = CHR$ (96) THEN
       A$ = ""
66 63200 RETURN
```

Program 2: MOUSEY Filemaker

```
B4 10 REM BASIC PROGRAM FDR
7C 20 REM GENERATING THE
4A 30 REM BINARY FILE
9A 40 REM 'MOUSEY'
4E 50 HOME
36 60 VTAB 12: PRINT "WORKING ..
    ."
21 70 FOR I = 0 TO 872
8B 80 REAO A
19 90 POKE 37376 + I,A
BF 100 VTAB 12: HTAB 13: PRINT I
     + 1
DF 110 NEXT I
2A 120 PRINT CHR$ (4)"BSAVE MOUS
     EY,A37376, L873"
2B 130 PRINT : PRINT "OONE!"
77 10000 OATA 173,89,170,72,165,
     217,72
98 10010 OATA 165,118,72,169,2,1
     33,118
B4 10020 OATA 169,255,133,217,16
     9,191,133
B7 10030 DATA 51,169,0,133,243,7
     6,41
6E 10040 OATA 146,4,0,15,0,7,0
AB 10050 OATA 21,0,7,0,21,0,169
D6 10060 DATA 29,133,133,169,146
     ,160,0
B4 10070 OATA 162,11,32,77,149,7
     6,69
57 10080 OATA 146,20,7,32,0,8,0
1B 10090 OATA 22,0,8,0,22,0,169
AB 10100 DATA 57,133,133,169,146
     ,160,0
35 10110 OATA 162,11,32,77,149,1
     73,0
FE 10120 OATA 3,141,29,146,169,0
     ,141
FE 10130 DATA 30,146,173,1,3,141
     ,31
B1 10140 OATA 146,169,0,141,32,1
     46,173
F8 10150 OATA 2,3,141,33,146,169
     ,0
6B 10160 OATA 141,34,146,173,3,3
     ,141
1D 10170 OATA 35,146,169,0,141,3
     6,146
D7 10180 OATA 173,4,3,141,37,146
     ,169
```

```
5C 10190 DATA Ø,141,38,146,173,5
     ,3
2C 10200 DATA 141,39,146,169,Ø,1
     41,4Ø
B3 10210 DATA 146,173,29,146,141
     ,61,146
C8 10220 DATA 173,3Ø,146,141,62,
     146,173
1B 10230 DATA 62,146,2Ø5,34,146,
     48,15
15 10240 DATA 2Ø8,1Ø,173,61,146,
     2Ø5,33
22 10250 DATA 146,144,5,24Ø,3,76
     ,134
D5 10260 DATA 147,173,31,146,141
     ,63,146
75 10270 DATA 173,32,146,141,64,
     146,173
BB 10280 DATA 64,146,2Ø5,36,146,
     48,15
7F 10290 DATA 2Ø8,1Ø,173,63,146,
     2Ø5,35
9D 10300 DATA 146,144,5,24Ø,3,76
     ,123
FB 10310 DATA 147,24,173,37,146,
     1Ø5,1
39 10320 DATA 141,65,146,173,38,
     146,1Ø5
2A 10330 DATA Ø,141,66,146,24,17
     3,39
BE 10340 DATA 146,1Ø5,1,141,67,1
     46,173
B3 10350 DATA 4Ø,146,1Ø5,Ø,141,6
     8,146
B1 10360 DATA 173,62,146,2Ø5,66,
     146,48
43 10370 DATA 1Ø,2Ø8,29,173,61,1
     46,2Ø5
F3 10380 DATA 65,146,176,21,173,
     64,146
6E 10390 DATA 2Ø5,68,146,48,1Ø,2
     Ø8,11
B2 10400 DATA 173,63,146,2Ø5,67,
     146,176
9C 10410 DATA 3,76,112,147,32,12
     2,148
EF 10420 DATA 173,57,146,141,51,
     147,173
A2 10430 DATA 58,146,141,52,147,
     173,144
A4 10440 DATA 7,141,59,146,169,Ø
     ,141
29 10450 DATA 6Ø,146,173,6Ø,146,
     2Ø1,Ø
D2 10460 DATA 48,9,2Ø8,24,173,59
     ,146
9B 10470 DATA 2Ø1,128,176,17,24,
     173,59
91 10480 DATA 146,1Ø5,128,141,59
     ,146,173
EA 10490 DATA 6Ø,146,1Ø5,Ø,141,6
     Ø,146
4B 10500 DATA 173,57,146,141,11Ø
     ,147,173
5A 10510 DATA 58,146,141,111,147
     ,173,59
5D 10520 DATA 146,141,144,7,238,
     63,146
9F 10530 DATA 2Ø8,3,238,64,146,7
     6,195
8E 10540 DATA 146,238,61,146,2Ø8
     ,3,238
AB 10550 DATA 62,146,76,16Ø,146,
     173,29
9B 10560 DATA 146,141,61,146,173
     ,3Ø,146
2B 10570 DATA 141,62,146,173,62,
     146,2Ø5
DB 10580 DATA 38,146,48,15,2Ø8,1
     Ø,173
1B 10590 DATA 61,146,2Ø5,37,146,
     144,5
43 10600 DATA 24Ø,3,76,119,148,1
     73,31
BF 10610 DATA 146,141,63,146,173
     ,32,146
BB 10620 DATA 141,64,146,173,64,
     146,2Ø5
96 10630 DATA 4Ø,146,48,15,2Ø8,1
     Ø,173
3C 10640 DATA 63,146,2Ø5,39,146,
     144,5
A3 10650 DATA 24Ø,3,76,1Ø8,148,3
     2,122
FF 10660 DATA 148,173,57,146,141
     ,22Ø,147
CB 10670 DATA 173,58,146,141,221
     ,147,173
2C 10680 DATA 2Ø,7,141,59,146,16
     9,Ø
A7 10690 DATA 141,6Ø,146,173,6Ø,
     146,2Ø1
57 10700 DATA Ø,48,28,2Ø8,9,173,
     59
B4 10710 DATA 146,2Ø1,127,144,19
     ,24Ø,17
B5 10720 DATA 56,173,59,146,233,
     64,141
B7 10730 DATA 59,146,173,6Ø,146,
     233,Ø
B3 10740 DATA 141,6Ø,146,173,6Ø,
     146,2Ø1
7F 10750 DATA Ø,48,28,2Ø8,9,173,
     59
DC 10760 DATA 146,2Ø1,64,144,19,
     24Ø,17
DD 10770 DATA 56,173,59,146,233,
     64,141
DF 10780 DATA 59,146,173,6Ø,146,
     233,Ø
AB 10790 DATA 141,6Ø,146,173,6Ø,
     146,2Ø1
5B 10800 DATA Ø,48,28,2Ø8,9,173,
     59
E7 10810 DATA 146,2Ø1,64,144,19,
     24Ø,17
B9 10820 DATA 56,173,59,146,233,
     64,141
BB 10830 DATA 59,146,173,6Ø,146,
     233,Ø
1B 10840 DATA 141,6Ø,146,173,57,
     146,141
3B 10850 DATA 95,148,173,58,146,
     141,96
5A 10860 DATA 148,173,59,146,141
     ,2Ø,7
92 10870 DATA 238,63,146,2Ø8,3,2
     38,64
1D 10880 DATA 146,76,181,147,238
     ,61,146
FA 10890 DATA 2Ø8,3,238,62,146,7
     6,146
43 10900 DATA 147,76,11,149,56,1
     73,61
ED 10910 DATA 146,233,1,141,57,1
     46,173
0A 10920 DATA 62,146,233,Ø,141,5
     8,146
4C 10930 DATA 169,Ø,133,138,169,
     128,174
E9 10940 DATA 58,146,172,57,146,
     32,31
45 10950 DATA 149,142,58,146,14Ø
     ,57,146
3B 10960 DATA 24,169,255,1Ø9,57,
     146,141
97 10970 DATA 57,146,169,3,1Ø9,5
     8,146
FB 10980 DATA 141,58,146,24,173,
     57,146
5B 10990 DATA 1Ø9,63,146,141,57,
     146,173
BF 11000 DATA 58,146,1Ø9,64,146,
     141,58
C4 11010 DATA 146,173,62,146,2Ø1
     ,Ø,48
CE 11020 DATA 28,2Ø8,9,173,61,14
     6,2Ø1
11 11030 DATA 8,144,19,24Ø,17,56
     ,173
96 11040 DATA 57,146,233,216,141
     ,57,146
DF 11050 DATA 173,58,146,233,3,1
     41,58
EC 11060 DATA 146,173,62,146,2Ø1
     ,Ø,48
F6 11070 DATA 28,2Ø8,9,173,61,14
     6,2Ø1
EF 11080 DATA 16,144,19,24Ø,17,5
     6,173
BE 11090 DATA 57,146,233,216,141
     ,57,146
B9 11100 DATA 173,58,146,233,3,1
     41,58
B5 11110 DATA 146,96,1Ø4,133,118
     ,1Ø4,133
1C 11120 DATA 217,1Ø4,141,89,17Ø
     ,169,141
67 11130 DATA 141,1,2,169,1,133,
     52
9A 11140 DATA 96,133,137,132,135
     ,134,136
BB 11150 DATA 169,Ø,133,133,133,
     134,7Ø
B1 11160 DATA 136,1Ø2,135,144,13
     ,24,165
BE 11170 DATA 137,1Ø1,133,133,13
     3,165,138
EC 11180 DATA 1Ø1,134,133,134,6,
     137,38
B5 11190 DATA 138,165,136,5,135,
     2Ø8,227
5B 11200 DATA 164,133,166,134,96
     ,133,134
FE 11210 DATA 132,135,16Ø,Ø,169,
     Ø,145
BB 11220 DATA 133,2ØØ,2Ø8,2,23Ø,
     134,138
2C 11230 DATA 2Ø8,4,198,135,48,4
     ,2Ø2
5B 11240 DATA 76,83,149,96,Ø,Ø
```

Program 3: TEXT Filemaker

```
51 1Ø D$ = CHR$ (4)
B7 2Ø PRINT D$"DPEN TEXT"
CF 3Ø PRINT D$"WRITE TEXT"
EA 4Ø PRINT "CDMPUTE!"
7E 5Ø PRINT "READER SERVICE"
89 6Ø PRINT "P.D. 8DX 5Ø95Ø
E3 7Ø PRINT "DES MDINES"
6A 8Ø PRINT "IA 5Ø95Ø"
94 9Ø PRINT ""
E9 1ØØ PRINT "1-8ØØ-346-6767"
DA 11Ø PRINT D$"CLDSE"
```

Program 4: Joystick Modifications

```
FC 265 8Ø = 4
CB 1Ø15Ø XØ = PDL (Ø)
B4 1Ø16Ø YØ = PDL (1)
7B 1Ø161 81 = PEEK ( - 16287)
21 1Ø162 IF 81 < 128 AND 8Ø = 3
      THEN BØ = 4
21 1Ø163 IF 81 < 128 AND 8Ø = 2
      THEN 8Ø = 3
29 1Ø164 IF B1 < 128 AND 8Ø = 1
      THEN 8Ø = 3
25 1Ø165 IF 81 > 127 AND 8Ø = 2
      THEN 8Ø = 1
4D 1Ø166 IF 81 > 127 AND 8Ø = 4
      THEN 8Ø = 2
21 1Ø17Ø IF PEEK ( - 16384) > 12
      7 THEN 1Ø44Ø
A9 1Ø18Ø YØ = INT (YØ / 1Ø) + 1
BB 1Ø19Ø XØ = INT (XØ / 6) + 1
32 2ØØ3Ø REM
```

©

IBM Variable Snapshot

Tony Roberts, Production Director

This programming utility lets you list the current values of all active variables in any BASIC program—an invaluable aid for debugging. It works on any IBM PC with BASICA or PCjr with Cartridge BASIC.

When things go haywire with a BASIC program, my first inclination is to check the variables: PRINT A$, PRINT SCORE, PRINT UPPERLIMIT, and so on. Comparing what's actually stored in a variable with what you expected often helps to isolate programming problems.

Printing variable values over and over, however, quickly becomes tedious, especially when arrays are involved. "IBM Variable Snapshot" takes the work out of this process.

After temporarily appending the Variable Snapshot utility to your program, you can activate it with a simple GOTO command whenever your program stops with an error or you press the BREAK key. Once activated, Variable Snapshot sifts through memory, printing out first the scalar variables, then the array variables it finds there. Within seconds, you can see the values of all the variables your program has used. This kind of analysis has many benefits:

- By frequently checking the variable list, you reduce the possibility of "forgotten" variables.

- You can quickly spot typographical errors in variable names. If the list contains both FILENAME$ and FILENAM$, you'll realize something is wrong.

- By checking variable types as well as names, you'll notice if the list contains both TOTAL% (an integer variable) and TOTAL! (a single-precision variable)—another common source of errors.

How To Take Snapshots

Type in Program 1 below and save it on disk in ASCII format. If you type it in with the "IBM Automatic Proofreader," published elsewhere in this issue, the program is saved in ASCII format automatically. Otherwise, use the command SAVE "SNAPSHOT.ASC",A.

Program 2 lets you test Variable Snapshot to verify that it's working properly before using it with your own programs. To run a test, type in Program 2 and save it on disk in ASCII format. Then append Snapshot to it with the command MERGE "SNAPSHOT.ASC." Now type RUN. The test program initializes several variables, then stops. When you type GOTO 1000 (the starting line number of Variable Snapshot), the name and value of each variable is printed on the screen. You can press CTRL-NUM LOCK on the PC or Fn-Q on the PCjr to pause the display, or stop it at any time by pressing CTRL-BREAK on the PC or Fn-BREAK on the PCjr.

If the variable values are not what you expected, recheck your typing, paying close attention to the type declaration symbols (%, $, !, #) attached to the variables. If even one of these symbols is incorrect or missing, you'll have problems.

The test program initializes integer, string, single-precision, and double-precision variables as well as a full set of array variables. If everything prints out as expected, you can be pretty sure that Variable Snapshot is working well.

Friendly Filename And Quick Start

When Snapshot begins its work, the first thing it prints is the active disk filename, which the IBM stores in the 11 memory locations beginning at 4F1h (1265 decimal). This has nothing at all to do with variables, but simply provides an answer to the question "What did I call this program the last time I saved it?"

If you want to get started with Snapshot quickly, you can omit the entire array processing section (lines 1590–2220) and change line 1280 to read:

```
1280 IF QARRAYON% THEN END
```

This abbreviated version of Snapshot lists only simple variables, but you can go back later and add the lines to handle the array

Variable	Description
Q%,QQ%,QQQ%	loop counters
QTYPE%	variable type
QLENLEFT%	number of characters left in variable name
QDIMS%	number of dimensions in array
QARRAYON%	flag indicating if array boundary passed
QSTRLEN%	length of string variable
QBASE%	status of OPTION BASE command
Q$	for single- and double-precision conversions
QCHAR$	builds active filename
QFILE$	active filename
QNAME$	name of variable being processed
QVAR!	memory pointer to current variable
QARRAY!	start of array space
QFREE!	start of free space
QASIZE!	size of current array
QVALUE!	temporary storage for integer values
QSTRPTR!	points to location of actual string
QPTR!	points to start of next element in array
QDIMSIZE( )	size of array dimensions

variables. The REMs in the program listing are not referenced by other lines, so you can safely omit them when typing the program.

After you have Snapshot working, edit line 1000 to suit your preferences for screen color, width, and so on.

You may want to renumber Snapshot so its line numbers won't interfere with those of your own programs. (Low line numbers were used in the listing to make entering the program easier.) Load the program into memory and use the command RENUM *xxxxx*, where *xxxxx* is Snapshot's new starting line number. Then save the program back to disk, again using the ASCII option so Snapshot can be merged other programs.

The version I use begins at line 60000, and I've programmed a function key to execute the command GOTO 60000. Whenever a program halts, I simply press Fn-6 to see the value of every variable.

Array Bases

IBM BASIC includes the OPTION BASE statement for defining the lowest-numbered element in an array. If a program contains the statement OPTION BASE 0, or if no OPTION BASE statement is included, all arrays start with a 0 element. An OPTION BASE 1 statement means that arrays begin with element 1.

Variable Snapshot must know which OPTION BASE is in effect to display array values properly. Memory location 45Ch (1116 decimal) provides this information. PEEKing that address yields either a 0 or 1, indicating which base is selected.

The adjacent memory location, 45Dh (1117 decimal), is related but a little more specific. If no OPTION BASE command has been issued, 45Dh contains a 0; if OPTION BASE 0 has been executed, 45Dh contains a 1; and if OPTION BASE 1 has been executed, the location contains a 2.

Try changing line 10 in Program 2 to read OPTION BASE 0 and observe the effect when running Variable Snapshot.

Although IBM BASIC allows arrays of up to 255 dimensions, few programs make use of more than one or two. For this reason, Variable Snapshot does not include provisions for arrays with more than two dimensions. Additional loops can be added to handle more complex arrays, if necessary.

A Few Cautions

To be truthful, Snapshot does not list *every* variable—it ignores those that begin with the letter Q. The Snapshot routine itself, you'll notice, uses only variables beginning with the letter Q. That keeps Snapshot's own variables from being printed along with those of your program.

If you're inclined to tinker with this routine, you must be careful about introducing new variables. Lines 1020–1040 initialize every variable used by the routine, effectively reserving space for them in the variable table.

Lines 1120–1140 determine the boundaries of the variable table, reference points the program cannot do without. If a new variable is added to the program after the boundary measurements are taken, confusion results; the boundaries move and Snapshot loses its way.

Although Snapshot works with most programs, there can be complications. If you've written your program to make use of all available memory, there won't be room in the variable table for Snapshot's own variables. You'll need to leave Snapshot about 300 bytes of workspace.

How Snapshot Works

As mentioned above, Snapshot reads the boundaries of the scalar variable area, the array variable area, and the free space area, then works its way through the variable areas byte by byte deciphering the information stored there. Once it reaches free space, its work is finished.

The IBM stores scalar variables as shown below.

Following the last character of the variable name is the value of the variable.

- An integer variable is stored in two bytes in the standard low byte/high byte format. The high bit of the second byte indicates the sign of the integer. If it is set, the integer is a negative number.

Byte 1 = type (2 = integer, 3 = string, 4 = single precision, 8 = double precision)
Byte 2 = first character of variable name
Byte 3 = second character of variable name
Byte 4 = number of characters remaining in variable name
Byte 5

.
. = rest of variable name (high bit set)
.

• String variable pointers are stored in three bytes. The first is the number of bytes in the string, and the second and third point to the address (either in the string pool or in the BASIC program area) where the string is stored.

• Single-precision variable values are stored in four bytes. The values of these bytes can be concatenated into a string, then converted into a single-precision number using the CVS function.

• Double-precision variables occupy eight bytes, which can be concatenated and converted as above using the CVD function.

Array variables are stored similarly, but there's some additional information between the end of the variable name and the actual beginning of the variable values.

Following the variable name are two bytes that indicate the total size of the array. The next byte holds the number of dimensions. That is followed by two bytes describing the number of elements in the last dimension. Then two bytes describe the number of elements in the next to last dimension, and so on, until each dimension in the array has been defined.

Finally, the values of the array variables follow, and are stored in the same manner as values for scalar variables.

Using this information, the program listing, the description of Snapshot variables found in the accompanying table, and the actual program output, you should be able to develop a good understanding of how BASIC treats your variables.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

Program 1: IBM Variable Snapshot

```
CK 1ØØØ DEF SEG:SCREEN Ø,Ø:WIDTH 8Ø:CO
        LOR 7,Ø:
OB 1Ø1Ø REM initialize variables
CG 1Ø2Ø Q%=Ø:QQ%=Ø:QQQ%=Ø:QLENLEFT%=Ø:
        QTYPE%=Ø:QD1MS%=Ø:QARRAYON%=Ø:
        QSTRLEN%=Ø:QBASE%=Ø:QD1MS1ZE%(
        1)=Ø:QDIMSIZE(2)=Ø
LM 1Ø3Ø Q$="":QCHAR$="":QFILE$="":QNAM
        E$=""
MJ 1Ø4Ø QVAR'=Ø:QARRAY'=Ø:QFREE'=Ø:QAS
        IZE'=Ø:QVALUE'=Ø:QSTRPTR'=Ø:QP
        TR'=Ø
GG 1Ø5Ø REM Get active filename
DL 1Ø6Ø FOR Q%=Ø TO 1Ø
BA 1Ø7Ø QCHAR$=CHR$(PEEK(&H4F1+Q%))
PA 1Ø8Ø IF ASC(QCHAR$)>96 AND ASC(QCHA
        R$)<123 THEN QFILE$=QFILE$+CHR
        $(ASC(QCHAR$)-32) ELSE QFILE$=
        QFILE$+QCHAR$
AH 1Ø9Ø NEXT
LI 11ØØ PRINT:PRINT "Active disk filen
        ame is:  ";M1D$(QFILE$,1,8);".
        ";MID$(QFILE$,9):PR1NT
CM 111Ø REM get addresses of scalar va
        riables, array variables, and
        free space
FK 112Ø QVAR'=PEEK(&H358)+PEEK(&H359)*
        256
HI 113Ø QARRAY'=PEEK(&H35A)+PEEK(&H35B
        )*256
HO 114Ø QFREE'=PEEK(&H35C)+PEEK(&H35D)
        *256
HF 115Ø QBASE%=PEEK(&H45C)
BN 116Ø REM Start of variable processi
        ng
CH 117Ø QTYPE%=PEEK(QVAR')
JO 118Ø IF (QTYPE%<2 OR QTYPE%>4) AND
        QTYPE%<>8 THEN END
LB 119Ø QLENLEFT%=PEEK(QVAR'+3)
PH 12ØØ REM get variable name
QN 121Ø QNAME$=""
LL 122Ø IF PEEK(QVAR'+1)>127 OR (PEEK(
        QVAR'+1)=81 AND QARRAYON%=Ø) T
        HEN 224Ø
EC 123Ø FOR Q%=1 TO QLENLEFT%
HJ 124Ø QNAME$=QNAME$+CHR$(PEEK(QVAR'+
        3+Q%) AND 127)
QP 125Ø NEXT
NE 126Ø QNAME$=CHR$(PEEK(QVAR'+1))+CHR
        $(PEEK(QVAR'+2))+QNAME$
OI 127Ø REM branch to appropriate rout
        ine depending on variable type
EG 128Ø IF QARRAYON% THEN 16ØØ
CF 129Ø ON QTYPE%-1 GOTO 132Ø,137Ø,146
        Ø
MP 13ØØ GOTO 153Ø
FK 131Ø REM inteters
EC 132Ø QVALUE'=PEEK(QVAR'+QLENLEFT%+4
        )+PEEK(QVAR'+QLENLEFT%+5)*256
DP 133Ø IF QVALUE'>32768' THEN QVALUE'
        =QVALUE'-65536'
PH 134Ø PRINT QNAME$;"%";,"= ";QVALUE'
MC 135Ø GOTO 224Ø
FE 136Ø REM strings
PH 137Ø PRINT QNAME$;"$","=  ";CHR$(34
        );
BN 138Ø QSTRLEN%=PEEK(QVAR'+QLENLEFT%+
        4)
MC 139Ø QSTRPTR'=PEEK(QVAR'+QLENLEFT%+
        5)+PEEK(QVAR'+QLENLEFT%+6)*256
GJ 14ØØ FOR Q%=Ø TO QSTRLEN%-1
PM 141Ø PR1NT CHR$(PEEK(QSTRPTR'+Q%));
QK 142Ø NEXT
GI 143Ø PR1NT CHR$(34)
MB 144Ø GOTO 224Ø
PJ 145Ø REM single precision
PL 146Ø Q$=""
FB 147Ø PRINT QNAME$;"'","= ";
CM 148Ø FOR Q%=Ø TO 3: Q$=Q$+CHR$(PEEK
        (QVAR'+QLENLEFT%+4+Q%))
BP 149Ø NEXT
BE 15ØØ PRINT CVS(Q$)
NK 151Ø GOTO 224Ø
KB 152Ø REM double precision
OE 153Ø Q$=""
HH 154Ø PRINT QNAME$;"#","= ";
BN 155Ø FOR Q%=Ø TO 7: Q$=Q$+CHR$(PEEK
        (QVAR'+QLENLEFT%+4+Q%))
BI 156Ø NEXT
FE 157Ø PRINT CVD(Q$)
OP 158Ø GOTO 224Ø
AC 159Ø REM array routines
JM 16ØØ QASIZE'=PEEK(QVAR'+4+QLENLEFT%
        )+PEEK(QVAR'+5+QLENLEFT%)*256
JI 161Ø IF ASC(QNAME$)=81 THEN 224Ø
BA 162Ø QDIMS%=PEEK(QVAR'+6+QLENLEFT%)
IB 163Ø IF QDIMS%>2 THEN 224Ø
QL 164Ø QPTR'=QVAR'+7+QLENLEFT%
MA 165Ø FOR Q%=QDIMS% TO 1 STEP -1
LN 166Ø QDIMS1ZE%(Q%)=PEEK(QPTR')+PEEK
        (QPTR'+1)*256
NE 167Ø QPTR'=QPTR'+2
BA 168Ø NEXT
JG 169Ø ON QTYPE%-1 GOTO 172Ø,183Ø,198
        Ø
KI 17ØØ GOTO 211Ø
QL 171Ø REM integer arrays
AB 172Ø PR1NT
BL 173Ø IF QDIMS%=2 THEN FOR QQQ%=QBAS
        E% TO QOIMSIZE%(2)+(QBASE%=Ø)
GI 174Ø FOR Q%=QBASE% TO QDIMSIZE%(1)+
        (QBASE%=Ø)
ON 175Ø QVALUE'=PEEK(QPTR')+PEEK(QPTR'
        +1)*256
AA 176Ø IF QVALUE'>32768' THEN QVALUE'
        =QVALUE'=65536!
IF 177Ø IF QOIMS%=1 THEN PRINT QNAME$;
        "%(";MID$(STR$(Q%),2);")","= "
        ;QVALUE' ELSE PRINT QNAME$;"%(
        ";MID$(STR$(Q%),2);",";MID$(ST
        R$(QQQ%),2);")","= ";QVALUE'
OJ 178Ø QPTR'=QPTR'+2
JM 179Ø NEXT Q%
EH 18ØØ IF QOIMS%=2 THEN NEXT QQQ%
MA 181Ø GOTO 224Ø
HA 182Ø REM string arrays
AB 183Ø PRINT
BA 184Ø IF QOIMS%=2 THEN FOR QQQ%=QBAS
        E% TO QDIMSIZE%(2)+(QBASE%=Ø)
GM 185Ø FOR Q%=QBASE% TO QOIMSIZE%(1)+
        (QBASE%=Ø)
IE 186Ø QSTRLEN%=PEEK(QPTR')
KA 187Ø QSTRPTR'=PEEK(QPTR!+1)+PEEK(QP
        TR'+2)*256
CO 188Ø IF QOIMS%=1 THEN PRINT QNAME$;
        "$(";MID$(STR$(Q%),2);")","= "
        ;CHR$(34); ELSE PRINT QNAME$;"
        $(";MID$(STR$(Q%),2);",";MID$(
        STR$(QQQ%),2);")","= ";CHR$(34
        );
CI 189Ø FOR QQ%=Ø TO QSTRLEN%-1
HF 19ØØ PR1NT CHR$(PEEK(QSTRPTR'+QQ%))
        ;
HJ 191Ø NEXT QQ%
BP 192Ø PRINT CHR$(34)
QA 193Ø QPTR'=QPTR'+3
IB 194Ø NEXT Q%
BI 195Ø IF QOIMS%=2 THEN NEXT QQQ%
OB 196Ø GOTO 224Ø
EL 197Ø REM single precision arrays
BH 198Ø PRINT
CB 199Ø IF QDIMS%=2 THEN FOR QQQ%=QBAS
        E% TO QOIMSIZE%(2)+(QBASE%=Ø)
EP 2ØØØ FOR Q%=QBASE% TO QOIMSIZE%(1)+
        (QBASE%=Ø)
NF 2Ø1Ø Q$=""
IN 2Ø2Ø FOR QQ%=Ø TO 3
CB 2Ø3Ø Q$=Q$+CHR$(PEEK(QPTR!+QQ%))
BB 2Ø4Ø NEXT QQ%
AH 2Ø5Ø IF QO1MS%=1 THEN PRINT QNAME$;
        "!(";MID$(STR$(Q%),2);")","= "
        ;CVS(Q$) ELSE PRINT QNAME$;"!(
        ";M1D$(STR$(Q%),2);","MID$(STR
        $(QQQ%),2);")","= ";CVS(Q$)
PK 2Ø6Ø QPTR!=QPTR'+4
IJ 2Ø7Ø NEXT Q%
FA 2Ø8Ø 1F QOIMS%=2 THEN NEXT QQQ%
OJ 2Ø9Ø GOTO 224Ø
HB 21ØØ REM double precision arrays
PD 211Ø PRINT
QN 212Ø IF QOIMS%=2 THEN FOR QQQ%=QBA5
        E% TO QOIMSIZE%(2)+(QBASE%=Ø)
FK 213Ø FOR Q%=QBASE% TO QOIMSIZE%(1)+
        (QBASE%=Ø)
OA 214Ø Q$=""
OO 215Ø FOR QQ%=Ø TO 7
DM 216Ø Q$=Q$+CHR$(PEEK(QPTR!+QQ%))
HK 217Ø NEXT QQ%
CJ 218Ø IF QOIMS%=1 THEN PRINT QNAME$;
        "#(";MID$(STR$(Q%),2);")","= "
        ;CVD(Q$) ELSE PRINT QNAME$;"#(
        ";M1D$(STR$(Q%),2);",";MID$(ST
        R$(QQQ%),2);")","= ";CVD(Q$)
EK 219Ø QPTR'=QPTR'+8
HI 22ØØ NEXT Q%
EP 221Ø 1F QDIMS%=2 THEN NEXT QQQ%
HI 222Ø GOTO 224Ø
BH 223Ø REM Get address of next variab
        le
MM 224Ø IF QARRAYON%<>1 THEN QVAR'=QVA
        R!+QLENLEFT%+QTYPE%+4 ELSE QVA
        R!=QVAR'+QASIZE'+QLENLEFT%+6
HK 225Ø 1F QVAR'=>QARRAY' THEN QARRAYO
        N%=1
MG 226Ø IF QVAR'=>QFREE' THEN END
CN 227Ø GOTO 117Ø
```

Program 2: Snapshot Demo

```
BL 1Ø REM Snapshot demo program
GK 2Ø OPTION BASE 1
LF 3Ø A%=2:A$="This is a string.":A'=1
      ØØØØ1':A#=345692811#
LA 4Ø DIM INTEGER%(5),STRIN$(5),SINGLE
      '(5),DOUBLE#(5)
BM 5Ø D1M IGR%(5,3),STN$(5,3),SNG!(5,3
      ),DBL#(5,3)
IJ 6Ø FOR I=1 TO 5:INTEGER%(I)=I:STRIN
      $(I)=CHR$(64+I):SINGLE'(I)=I*3ØØ
      ØØ:DOUBLE#(I)=I*1.5E+Ø7:NEXT I
MG 7Ø FOR I=1 TO 5:FOR J=1 TO 3:IGR%(I
      ,J)=I+J:STN$(I,J)=CHR$(I+64)+CHR
      $(J+48):SNG'(I,J)=1ØØ*I*J:DBL#(I
      ,J)=I/J:NEXT J,I
OK 8Ø END
```

©

ATARI TEXTDUMP

Ralph Johnson

Here's a short, simple utility that quickly dumps a GRAPHICS 0 screen to a printer. It works with all 400/800, XL, and XE computers.

I've always wanted the capability to dump a copy of a text screen to my printer. I also wanted this capability available from BASIC. My efforts to find such a program were fruitless. So, the only solution was to write one myself.

There were several requirements I established: 1) It should be fast—written in machine language. 2) It should sit in a relatively safe location in memory, surviving system resets. 3) It should always be ready to do its job, whether called in direct mode or from within a program. 4) It should be easy to use.

The result is "Atari Textdump." You can make your own copy of Textdump by typing in the program listing below. It creates an AUTORUN.SYS file on disk that automatically loads Textdump into memory page 6 (address 1536) when you boot the system from that disk. To call the routine, make sure your printer is online and enter this statement:

A=USR(1536)

This works in both direct mode or within a program.

If you don't have a disk drive, or if you don't want Textdump to load as an AUTORUN.SYS file, delete lines 10–1000 in Program 1 and substitute this new line 10:

10 FOR A=1536 TO 1724:READ B:POKE A,B:NEXT A

Again, you can call Textdump as described above in either direct or program mode. You can also convert this version of the program into a module for use in your own programs.

If you like, you can modify Textdump to print a smaller portion of the GRAPHICS 0 screen. Simply POKE the desired number of rows you want to dump into memory location 1613.

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Atari Textdump

```
CL 10 CLOSE #1
PB 20 OPEN #1,8,0,"D:AUTORUN
      .SYS"
EJ 30 FOR A=1 TO 6:READ B:PR
      INT #1;CHR$(B);:NEXT A
IA 40 FOR A=1536 TO 1724:REA
      D B:PRINT #1;CHR$(B);:
      NEXT A
CP 50 CLOSE #1
CM 1000 DATA 255,255,0,6,188
        ,6
DG 1002 DATA 104,162,80,169,
        3,157,66,3,169,8,157
        ,74,3,169,144
IN 1003 DATA 157,68,3,169,6,
        157,69,3,169,0
GF 1004 DATA 141,143,6,157,7
        3,3,169,255,157,72,3
        ,32,86,228,165
KC 1005 DATA 88,133,203,165,
        89,133,204,162,0,160
BF 1006 DATA 0,24,177,203,10
        5,32,157,147,6,232,2
        4,165,203,105
EP 1007 DATA 1,133,203,165,2
        04,105,0,133,204,173
        ,143
OB 1008 DATA 6,201,24,240,52
        ,224,40,240,4,224,40
        ,208,217,169
DL 1009 DATA 155,157,147,6,2
        38,143,6,162,80,169,
        9
NP 1010 DATA 157,66,3,169,14
        7,157,68,3,169,6,157
        ,69,3,169,0
OL 1011 DATA 157,73,3,169,25
        5,157,72,3,32,86
EJ 1012 DATA 228,162,0,224,0
        ,240,173,162,80,169,
        12,157,66,3,32
BE 1013 DATA 86,228,96,0,80,
        58,155,0,0,0
LA 1014 DATA 0,0,0,0,0,0,0,0
        ,0,0,0,0,0,0,0,0,0,0
        ,0,0,0,0,0,0,0
CI 1015 DATA 0,0,0,0,0,0,0,0
        ,0,0,0,0,155,0
```

©

AmigaDOS Batch Files

Charles Brannon
Program Editor

AmigaDOS is more than a console-driven disk operating system. By executing a sequence of AmigaDOS commands stored in a file, AmigaDOS takes on some of the characteristics of a programming language. Whether you want to simplify repetitive disk commands or create personalized custom commands, batch files further extend the range and flexibility of AmigaDOS.

No matter how easy it is to use a program, the most popular programs are those that give users more power. And although a program may have scads of powerful commands, the most powerful programs are those which let users put the commands together in new ways—in effect, to write programs.

Instead of forcing you to always issue commands one at a time, a programmable application lets you create a script of commands to customize the behavior of the program. Whether we're talking about word processing macros, spreadsheet templates, relational database languages, or advanced machine language, programmability is the real key to software power. If you feel limited by a certain range of commands, you can combine the commands in new ways to create personalized features, just as we combine the vocabulary of English words to create a wealth of literature. Why just read when you can write?

Scripts, Sequences, And Batches

AmigaDOS is more than just a disk operating system—it's a programmable system that can process lists of its own commands as well as individual commands. In effect, AmigaDOS is a simple disk-oriented programming language.

A list of AmigaDOS commands can be stored in a disk file variously known as a *script*, a *sequence*, or a *batch file*. The term "batch file" is most commonly used by those who work with PC-DOS, MS-DOS, and CP/M, which are also programmable disk operating systems. To keep things straight, we'll use "batch files" synonymously with "scripts" or "sequences."

Even if you don't program in BASIC or any other language, you may be interested in learning about AmigaDOS batch files. The batch file "language" is simply made up of the same AmigaDOS commands you've probably been using all along (see "Introduction to AmigaDOS," Parts 1 and 2, in the February and March 1986 issues of COMPUTE!). There are also a few AmigaDOS commands designed especially for batch files.

Creating and running batch files is easy. Using a text editor, you just type in a list of AmigaDOS commands. Then you save the list on disk under a filename. To run the batch file, you type EXECUTE *filename* at an AmigaDOS prompt. AmigaDOS reads the batch file and executes the list of commands, just as if you had typed them one by one yourself.

We won't cover some of the more advanced features of batch files, useful only to advanced C and machine language programmers. Instead, we'll concentrate on the everyday utility of batch file programming.

A Quick Example

In a moment, we'll show how to create batch files with ED, the AmigaDOS full-screen text editor, but first there's a simpler way to create a short batch file. Enter this line at an AmigaDOS prompt:

```
copy * to Hello
```

(Note that AmigaDOS commands can be entered in uppercase or lowercase.)

Although nothing seems to happen, AmigaDOS is waiting for you to enter some lines. We'll use the ECHO command to display a friendly message. ECHO displays any text that follows it within quotation marks, just like the PRINT statement in BASIC. One difference is that if you want to ECHO only a single word, the quotes aren't necessary.

At an AmigaDOS prompt, enter the following text, pressing RETURN after each line:

```
echo "Hello!"
echo "I am your friend, the Amiga"
echo "personal computer."
```

After the last line, press CTRL-\. This key is the one to the left of the BACKSPACE key. CTRL-\ tells AmigaDOS that you're finished, and that it should finish writing and close the file. This key represents EOF, for End Of File.

To confirm that you've typed the file correctly, enter:

```
TYPE Hello
```

You should see the same lines you typed. Now you can start this simple program:

```
EXECUTE Hello
```

This should print on the screen:

```
Hello!
I am your friend, the Amiga
personal computer.
```

Using ED

It would be nice to have the Amiga actually speak this greeting. Rather than type in a whole new file, we'll use ED, the screen editor, to make the simple changes we're interested in. Enter:

```
ED Hello
```

This runs ED and also loads the batch file named Hello. When you start ED, you can give it the name of any file to edit. If the filename doesn't exist, it will be created; otherwise the file is automatically displayed on the editor screen. (Incidentally, AmigaDOS has another text editor called EDIT, but it's not as easy to use as ED.)

We'll make the Amiga speak the ECHO messages aloud by taking advantage of the system's built-in speech synthesis via the AmigaDOS SAY command (added to AmigaDOS version 1.1). To learn more about SAY, just enter SAY by itself to enter an interactive mode with on-screen instructions.

After you start ED by typing ED Hello, the batch file we previously entered should be on the screen, with the cursor at the beginning of the first line. ED is a full-screen text editor, so you can move the cursor anywhere within the file (but not past the last line). To insert some text, just start typing. The DEL and BACKSPACE keys can be used to delete characters.

Move the cursor to the second ECHO line and press RETURN. This inserts a blank line. Cursor up to the blank line and enter:

```
SAY HELLO!
```

You don't need to press RETURN at the end of the line, since you already did this to open up a line for typing.

Now cursor to the end of the file and type:

```
SAY I am your friend, the Amiga
personal computer.
```

(Notice that SAY is the only AmigaDOS command that doesn't require you to enclose text containing spaces with quotes.) This is how your screen should look:

```
echo "Hello!"
say Hello!
echo "I am your friend, the Amiga"
echo "personal computer."
say I am your friend, the Amiga personal
computer.
```

With the cursor at the end of the file, press the ESC key. An asterisk (*) should appear. Press the X key, then RETURN. This exits ED and saves your changes back to disk.

Finally, type EXECUTE Hello to try out your talking batch file.

Although these techniques are sufficient for simple editing, ED has dozens of editing commands. For example, CTRL-B (press CTRL and B at the same time) blanks out and deletes the line the cursor is on. ESC-J-RETURN joins two lines together. Space doesn't permit a discussion of all these commands, but if you like to experiment, refer to the abbreviated ED reference chart accompanying this article.

Startup-Sequence

A special AmigaDOS batch file, called the *startup-sequence*, is executed automatically when you boot up an AmigaDOS or Workbench disk by inserting it at the Workbench prompt. Startup-sequence normally just displays a message, then launches the Workbench and ends the command line interface.

To edit this batch file, enter:

```
ed s/startup-sequence
```

This runs ED and calls up the file "startup-sequence" from the S subdirectory. This subdirectory, which can also be accessed as the S: device, is a convenient place for batch files. Just as AmigaDOS by default searches for AmigaDOS commands in the C subdirectory, the EXECUTE command first looks for a batch file in the S subdirectory. If AmigaDOS can't find the batch file in this subdirectory, it looks for it in the current directory. So no matter what your current directory is, you can always use your batch file if you place it in the S directory on your startup disk.

When you first load startup-sequence into ED, you'll see something like this:

```
echo "Workbench disk. Release 1.1"
echo " "
echo "Use Preferences tool to set date"
echo " "
LoadWb
endcli > nil:
```

Since this message appears every time you start up your disk, you may want to change the ECHO statements for a personalized message. Likewise, if you'd rather use AmigaDOS instead of the Workbench, delete the last two lines. The "> nil:" sequence makes AmigaDOS throw away the output of a command; here, the message "CLI task 1 ending."

Startup-sequence is a good place to put personalized commands. For example, if you like to keep your command directory in RAM for speed and convenience, you could insert these lines above the LoadWb line:

```
makedir ram:c
copy c to ram:c all quiet
cd ram:c
```

This copies all of the AmigaDOS commands from the C subdirectory on the floppy disk into a C subdirectory on the RAM disk. It also changes the current directory to the C subdirectory in RAM:, so any AmigaDOS commands you type from then on will be loaded

Common ED Commands

Immediate Commands (hold down CTRL and press key):

CTRL-A	Insert line at cursor position
CTRL-B	Delete current line
CTRL-D	Scroll text downward
CTRL-E	Move cursor to top or bottom of screen
CTRL-N	Delete character at cursor
CTRL-O	Delete word or series of spaces
CTRL-U	Scroll text upward
CTRL-Y	Delete to end of current line

Extended Commands (precede by pressing and releasing ESC):

B	Move cursor to bottom of file
E/*string1*/*string2*/	Exchange *string1* to *string2*
EQ/*string1*/*string2*/	Exchange, but query first
F/*string*/	Find *string*
J	Join current line with next line
Q	Quit without saving text
T	Move cursor to top of file
X	Exit, save text

from RAM: instead of from the floppy. In effect, this turns AmigaDOS into a memory-resident DOS, with all commands intrinsic instead of extrinsic. AmigaDOS responds much faster this way. However, this also uses up quite a bit of memory, so you may want to copy only the commands you use frequently.

Another useful startup action is to set the date and time. You can always do this with the Preferences tool or by opening a CLI and using the DATE command. However, it can be more convenient to enter the date when you first turn on your Amiga, allowing all files subsequently saved to be stamped with the current date and time. Just insert this line into startup-sequence:

```
date ?
```

The ? operator can be used in place of the parameter of a command. Instead of specifying the date, ? prompts the user to enter the date. It also displays the template for the date command (TIME,DATE, TO=VER/K:). If you like, you can use ECHO to display your own prompt, and > nil: to discard the template:

```
echo "Please enter the date and time."
echo "DD-MMM-YY HH:MM:SS"
date > nil: ?
```

From then on, whenever you boot up from this disk, you'll respond to the prompt by typing something like this:

```
27-jan-86 15:12
```

which automatically sets the system clock.

Variable Parameters

You can also send special options to your batch file. You enter these options on the command line along with the EXECUTE command. Just as with variables in BASIC, you can manipulate these parameters symbolically.

Let's say you'd like a batch file that gives you complete information on a file. It uses LIST to display the information about the file, and TYPE to display the file. You would use a command like EXECUTE SHOW RODEO to display the file RODEO. Use ED SHOW or COPY * TO SHOW to create this batch file:

```
.KEY name
LIST <name>
TYPE <name>
```

.KEY (don't forget the leading period) sets up a name for substitution text. Whatever you typed on the same line with EXECUTE is substituted wherever you use <name>. You must use the angular brackets, or LIST and TYPE would look literally for the file "name."

After creating this batch file, type this at an AmigaDOS prompt:

```
EXECUTE SHOW S/STARTUP-
SEQUENCE
```

The result is the same as if you had typed LIST S/STARTUP-SEQUENCE followed by TYPE S/STARTUP-SEQUENCE.

Other AmigaDOS commands let you check to see if the user has entered a specific string and check to see if a file exists. To prevent an error message, we can check to see if the file exists before we use LIST and TYPE:

```
.KEY name
IF EXISTS <name>
LIST <name>
TYPE <name>
ELSE
ECHO "<name> does not exist!"
ENDIF
```

Notice the use of IF, ELSE, and ENDIF. Looks like Amiga BASIC, doesn't it? In fact, the AmigaDOS IF-ELSE-END IF commands function very much like BASIC's. When the IF condition is true, AmigaDOS executes the following statements; otherwise the following statements are ignored. ELSE executes the statements following it only if the preceding IF was false. ENDIF cancels conditional processing and returns to executing all commands.

Any Parameters Missing?

Here's how to use the IF EQ option to test for the existence of a command-line parameter. If there is no parameter, <name> is null, so "<name>z" is simply "z". We use NOT to reverse the test. If the parameter "<name>z" is NOT equal to "z", then we must have a command line parameter. (We can't just test IF <name> NOT EQ "", since EQ wants two parameters, and the null string "" is not a parameter, but the lack of one.)

```
.KEY name
IF <name>z NOT EQ z
LIST <name>
TYPE <name>
ELSE
ECHO "You didn't give me anything to
SHOW."
ENDIF
```

Although you can't use leading spaces in the actual batch file, it's easier to follow the IF-ENDIF structures when you use indentation. Just don't type in the leading spaces. This version of the batch file SHOW checks both for the existence of the filename and for the presence of the filename parameter:

```
.KEY name
IF <name>z NOT EQ z
    IF EXISTS <name>
        LIST <name>
        TYPE <name>
    ELSE
        ECHO "<name> does not exist!"
    ENDIF
ELSE
    ECHO "You didn't give me anything
    to SHOW."
ENDIF
```

You can use more than one parameter in the .KEY statement, just as many commands, such as DATE, accept two inputs.

If the user doesn't enter anything for the parameter, you can assign a default value using either .DEF or $. If you use .DEF, the default phrase is used throughout the batch file. In this example, SHOW displays itself if you don't give it a filename.

```
.KEY name
.DEF s/show
LIST <name>
TYPE <name>
```

You can use $ to substitute a default value only for the current substitution. Several batch commands may use the value in different ways, so each command may have its own default value. In the following example, LIST displays the whole directory if <name> is null, but TYPE types the file "TEMP" if <name> is null:

```
.KEY name
LIST <name>
TYPE <name$temp>
```

Labels And Branching

You can jump forward to a label with the SKIP command. You'd typically use SKIP along with an IF condition if you want to skip over a block of statements that shouldn't be executed if the IF was true. You declare the label with LAB. SKIP can't skip backward, only forward to a LAB statement. You can usually use IF and ELSE to accomplish the same thing, though.

```
.KEY name
IF exists <name>
TYPE <name>
SKIP ToMyLou
ENDIF
ECHO "<name> doesn't exist."
LAB ToMyLou
echo "Finished."
```

An EXECUTE command can execute another batch file, or even itself. This permits backward looping to some degree. Nested batch files can be quite handy. You can test and debug individual batch programs, then execute them together from a master execute script:

```
EXECUTE Greeting
EXECUTE GetDate
EXECUTE Assignments
```

The individual files could themselves contain other EXECUTE references.

ASSIGNing Shortcuts

If you're using EXECUTE a lot, you may grow weary of typing it. You can always rename EXECUTE to something short like x, but other batch programs may contain EXECUTE statements, requiring you to rename it back. Instead, you can use the ASSIGN command to assign any filename to a device name.

```
ASSIGN x: sys:c/EXECUTE
```

You can now use x: whenever you want to use the EXECUTE command. (The prefix *sys:c/* makes sure that EXECUTE can be found no matter what directory you're in.)

The device name you create should not conflict with an existing one. To get a list of the current assignments, just type ASSIGN. You may want to ASSIGN d: c:list for a convenient and quick shorthand for directories (c: is synonymous with the C directory). You can then just type d: to get a LIST.

ASSIGN can be so handy for this kind of thing that you'll probably want to include your own sequence of ASSIGN commands within startup-sequence. If you put your ASSIGN statement within startup-sequence, you'll get these assignments for every session. Just remember that ASSIGN can only be used to attach a device name to a particular filename. ASSIGN d: "c:list quick" doesn't seem to work. Although LIST is a filename in the c directory, the "quick" parameter is not part of the filename. ©

Apple Disk Duper

Jason Coleman

Here's a program that lets you duplicate Apple disks quickly and conveniently. Though it can copy disks formatted for either DOS 3.3 or ProDOS, it must be run with ProDOS. It also requires 128K RAM.

Everyone knows the value of backing up disks. But how many of us take the time to make archive copies of important disks on a regular basis? "Apple Disk Duper" simplifies the process by making it possible to copy an entire disk in only two passes. It works on one- or two-drive systems with at least 128K RAM.

After typing in the program and saving a copy, simply run it and follow the instructions on the screen. Apple Disk Duper prompts you every step of the way.

Although the program runs only under ProDOS, it can copy DOS 3.3 disks as well as ProDOS disks. It works with any Apple Disk II-compatible drive, but not with the new 3½-inch UniDisk.

Apple Disk Duper

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
74 1ØØ FOR X = 76B TO 7B5: READ
       Y: POKE X,Y: NEXT
AA 11Ø DATA 32,Ø,191,129,9,3,176
       ,249,96,3,96,Ø,32,Ø,Ø,Ø,Ø
       ,Ø,
5B 12Ø TEXT : HOME
47 13Ø VTAB 12: HTAB 12: PRINT "
       DISK DUPLICATOR"
AA 14Ø VTAB 2Ø: HTAB 9: PRINT "(
       HIT ANY KEY TO BEGIN)";:
       POKE - 16368,Ø: GET ST$
4E 15Ø HOME
1C 16Ø VTAB 12: INPUT "ENTER NUM
       BER OF DRIVES:";ND$:ND =
       VAL (ND$)
CB 17Ø IF ND < > 1 AND ND < > 2
       THEN 39Ø
F1 18Ø HOME : VTAB 12: PRINT "PU
       T SOURCE DISK IN DRIVE 1"
7A 19Ø IF ND = 2 THEN VTAB 17: P
       RINT "PUT DESTINATION DIS
       K IN DRIVE2"
D8 2ØØ VTAB 2Ø: POKE - 16368,Ø:
       PRINT "PRESS ANY KEY TO M
       AKE COPY.": GET AK$
9B 21Ø FB = Ø:MX = 3
91 22Ø FOR N = 1 TO MX
C4 23Ø POKE 771,128
11 24Ø POKE 78Ø,32: POKE 778,96
1E 25Ø FOR I = FB TO FB + 55
25 26Ø P2 = INT (I / 256):P1 = I
        - 256 * P2
DE 27Ø POKE 782,P2: POKE 781,P1
58 28Ø CALL 768: POKE 78Ø, PEEK
       (78Ø) + 2: NEXT I
84 29Ø IF N < MX THEN PRINT CHR$
        (4)"BSAVE/RAM/COPY"N",A$
       2ØØØ,L$6FFF":FB = FB + 56
62 3ØØ NEXT N
28 31Ø IF ND = 1 THEN VTAB 12: P
       RINT "PUT DESTINATION DIS
       K IN DRIVE 1": GET AK$
ED 32Ø FOR N = MX TO 1 STEP - 1
F8 33Ø POKE 771,129: POKE 78Ø,14
       2
15 34Ø IF ND = 2 THEN POKE 778,2
       24
68 35Ø IF N < MX THEN PRINT CHR$
        (4)"BLOAD/RAM/COPY";N
79 36Ø FOR I = FB + 55 TO FB STE
       P - 1:P2 = INT (I / 256):
       P1 = I - 256 * P2
DF 37Ø POKE 782,P2: POKE 781,P1
71 38Ø CALL 768: POKE 78Ø, PEEK
       (78Ø) - 2: NEXT I
8B 39Ø FB = FB - 56
63 4ØØ NEXT N
1B 41Ø IF MX = 2 THEN 44Ø
6D 42Ø MX = 2:FB = 168: IF ND =
       1 THEN VTAB 12: PRINT "PU
       T SOURCE DISK IN DRIVE 1
           ": GET AK$
16 43Ø GOTO 22Ø
87 44Ø HOME : VTAB 12: HTAB 15:
       INVERSE : PRINT "COPY COM
       PLETE": NORMAL : END
```
©

Smooth-Scrolling Billboards For IBM

Paul W. Carlson

Do you want to leave a message on your computer screen that's sure to be noticed? Or would you like to create an eye-catching display in a shop window that effectively communicates your message to the public? The programs presented here let you easily produce smooth-scrolling billboards on the 40- or 80-column screen of your IBM PC (with color/graphics adapter and BASICA) or PCjr (with Cartridge BASIC).

To be really effective, a billboard program must *smoothly* scroll its message across the screen. Programs that jerk the letters across the screen are very hard on the eyes. The speed necessary for smooth scrolling can be achieved only by avoiding the routines in the BIOS (Basic Input/Output System) and writing directly to video memory. However, this can cause a problem when text is used in graphics modes—writing directly to video memory disrupts the character generator. As a result, small flickering lines appear on the screen (for more details, see COMPUTE! Books' *Mapping the IBM PC and PCjr,* pages 193–198).

This problem can be solved by writing to video memory only during the time when the monitor's raster beam is in vertical retrace, while the display is idle. On some IBM-compatible computers (the Compaq, for example), the problem can be avoided by writing to an inactive page of video memory and then making it the active page. The programs following this article make use of both methods.

With some computer and graphics card combinations, a few flickering lines remain at the very top of the screen when running the 80-column billboard program. These could have been eliminated, but only at the expense of speed and smoothness. About 300 characters can be written to video memory during the vertical retrace period, and 640 characters (eight lines of 80) need to be written for each screen update. Therefore, to eliminate the flickering lines entirely, we'd have to wait for three vertical retrace periods. These lines are less objectionable than the loss of smoothness caused by waiting for an extra retrace period.

Creating Billboards

Program 1 is for creating billboards on the 40-column screen, and Program 2 is for the 80-column screen. Both programs are extremely easy to use. After typing RUN, simply enter any text string at the prompt. If you want your message to contain a comma, enclose the entire text string in double quotes. When you press ENTER, the message enlarges and begins scrolling. It can be stopped at any time by pressing the Q key.

The programs can be customized to suit your taste. The character that forms the large letters can be changed from a solid block to another character by changing the DATA statement identified in the listing. For example, to change the solid block to a smiling face, change the DB to 02 in line 300. You can also modify the scrolling speed by changing the two bytes identified in the listing (the second byte has 256 times the effect of the first byte).

How It Works

The techniques used here can be applied to any program that must update a text screen very rapidly, so a brief description of the steps involved may be useful.

1. Set up a buffer in memory equal in size to the block of characters to be written to the screen (8 × 80 for the 80-column billboard).

2. For each input character, access the character PEL map in ROM at FFA6:OE. By columns, depending on whether or not a bit is set, put the code for a solid block or a space into the rightmost column of the buffer array.

3. When a column is complete, scroll the whole buffer one column to the left.

4. Wait for the beginning of a vertical retrace period, then copy the buffer to the inactive screen.

5. Make the inactive screen the active screen.

6. Do the next column in step 2.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

Program 1: 40-Column Billboards

```
EM 10 '  Forty Column Scrolling
     8illboard
JO 20 '
LJ 30 '  Press the "Q" key to qu
     it.
JA 40 '
DI 5Ø DEF SEG:CLEAR,&H3FFØ:N=&H4
     6ØA
DL 6Ø FDR J=Ø TD 249:READ A$
BB 7Ø PDKE N+J,VAL("&H"+A$):NEXT
FA 8Ø KEY DFF:CLS:SCREEN Ø:W1DTH
      8Ø
GL 9Ø INPUT"Text string";T$:T$=T
     $+"  "
FF 1ØØ N=&H4ØØØ:K=LEN(T$):FDR J=
      1 TD K
LP 11Ø PDKE N,ASC(M1D$(T$,J,1)):
      N=N+1
OA 12Ø NEXT:POKE N,Ø:CLS:WIDTH 4
      Ø
JL 13Ø LDCATE,,Ø:N=&H46ØA:CALL N
CC 14Ø WIDTH 8Ø:CLS:KEY DN:END
EG 15Ø DATA Ø6,88,EC,8C,D8,8E,CØ
      ,89
EM 16Ø DATA 8Ø,Ø2,8D,3E,Ø8,41,1E
      ,88
KK 17Ø DATA ØØ,88,8E,D8,8E,3Ø,Ø2
      ,F3
BB 18Ø DATA A4,1F,88,A6,FF,8E,CØ
      ,8D
GG 19Ø DATA 36,ØØ,4Ø,8A,1C,46,8Ø
      ,F8
JK 2ØØ DATA ØØ,74,F4,87,ØØ,D1,E3
      ,D1
KI 21Ø DATA E3,D1,E3,83,C3,ØE,89
      ,Ø8
FK 22Ø DATA ØØ,33,FF,26,8A,Ø7,88
      ,85
GJ 23Ø DATA ØØ,41,47,43,E2,F5,56
      ,Ø6
DK 24Ø DATA 89,Ø9,ØØ,51,33,FF,89
      ,Ø8
FI 25Ø DATA ØØ,88,4E,ØØ,DØ,A5,ØØ
      ,41
MK 26Ø DATA 72,Ø4,8Ø,2Ø,E8,Ø2,8Ø
DF 27Ø ' The following value is
      the
JF 28Ø ' ASCII code of character
       that
HE 29Ø ' forms the large text.
HP 3ØØ DATA D8
JH 31Ø DATA B8,B7,ØB,41,B3,C3,5Ø
      ,47
DM 32Ø DATA E2,EA,EB,Ø2,E8,85,BC
      ,DB
CF 33Ø DATA 8E,C3,FC,8B,ØB,ØØ,8D
      ,36
EK 34Ø DATA ØA,41,BD,3E,Ø8,41,B9
      ,4E
FC 35Ø DATA ØØ,F3,A4,46,46,47,47
      ,4B
MF 36Ø DATA 75,F4,AØ,ØB,46,34,Ø1
      ,A2
HI 37Ø DATA Ø8,46,B4,Ø5,5Ø,AB,Ø1
      ,75
BO 38Ø DATA Ø5,BB,ØØ,BB,EB,Ø3,8B
      ,8Ø
GE 39Ø DATA 8B,BE,CØ,89,AØ,ØØ,BF
      ,3Ø
HK 4ØØ DATA Ø2,8D,36,ØB,41,8A,DA
      ,Ø3
OK 41Ø DATA EC,AB,ØB,75,FB,EC,AB
      ,ØB
GC 42Ø DATA 74,F8,F3,A5,E8,Ø4,E8
      ,88
JE 43Ø DATA E8,AA,89,AØ,ØØ,8A,DA
      ,Ø3
OA 44Ø DATA EC,A8,Ø8,75,F8,EC,A8
      ,Ø8
IJ 45Ø DATA 74,F8,F3,A5,5B,CD,1Ø
      ,89
FF 46Ø ' The following two value
      s are the
FJ 47Ø ' time delay constant in
      the order
KH 48Ø ' least sig. byte, most s
      ig. byte.
CP 49Ø DATA Ø1,ØØ
AG 5ØØ DATA E2,FE,59,E2,DF,Ø7,5E
      ,84
AG 51Ø DATA Ø6,82,FF,CD,21,3C,71
      ,74
MM 52Ø DATA Ø6,3C,51,74,Ø2,E8,CF
      ,88
FP 53Ø DATA E5,Ø7,88,ØØ,Ø5,CD,1Ø
      ,C8
```

Program 2: 80-Column Billboards

```
KF 1Ø '  Eighty Column Scrolling
      8illboard
JO 2Ø '
LJ 3Ø '  Press the "Q" key to qu
     it.
JA 4Ø '
DI 5Ø DEF SEG:CLEAR,&H3FFØ:N=&H4
     6ØA
HC 6Ø FOR J=Ø TD 25Ø:READ A$
BB 7Ø PDKE N+J,VAL("&H"+A$):NEXT
FA 8Ø KEY OFF:CLS:SCREEN Ø:WIDTH
      8Ø
GL 9Ø INPUT"Text string";T$:T$=T
     $+"  "
FF 1ØØ N=&H4ØØØ:K=LEN(T$):FDR J=
      1 TO K
LP 11Ø POKE N,ASC(MID$(T$,J,1)):
      N=N+1
HD 12Ø NEXT:PDKE N,Ø:CLS
JL 13Ø LDCATE,,Ø:N=&H46ØA:CALL N
CC 14Ø WIDTH 8Ø:CLS:KEY DN:END
EG 15Ø DATA Ø6,8B,EC,8C,D8,8E,CØ
      ,89
BH 16Ø DATA ØØ,Ø5,8D,3E,Ø8,41,1E
      ,88
CP 17Ø DATA ØØ,88,8E,D8,8E,6Ø,Ø4
      ,F3
BB 18Ø DATA A4,1F,88,A6,FF,8E,CØ
      ,8D
GG 19Ø DATA 36,ØØ,4Ø,8A,1C,46,8Ø
      ,F8
JK 2ØØ DATA ØØ,74,F4,87,ØØ,D1,E3
      ,D1
KI 21Ø DATA E3,D1,E3,B3,C3,ØE,89
      ,Ø8
FK 22Ø DATA ØØ,33,FF,26,BA,Ø7,BB
      ,85
GJ 23Ø DATA ØØ,41,47,43,E2,F5,56
      ,Ø6
DK 24Ø DATA 89,Ø9,ØØ,51,33,FF,B9
      ,Ø8
KI 25Ø DATA ØØ,BB,9E,ØØ,DØ,A5,ØØ
      ,41
MK 26Ø DATA 72,Ø4,BØ,2Ø,E8,Ø2,BØ
DF 27Ø ' The following value is
      the
JF 28Ø ' ASCII code of character
       that
HE 29Ø ' forms the large text.
HP 3ØØ DATA D8
GN 31Ø DATA 88,B7,ØB,41,81,C3,AØ
      ,ØØ
JO 32Ø DATA 47,E2,E9,E8,Ø2,E8,B4
      ,8C
BE 33Ø DATA DB,BE,C3,FC,BB,Ø8,ØØ
      ,BD
ME 34Ø DATA 36,ØA,41,8D,3E,Ø8,41
      ,89
CP 35Ø DATA 9E,ØØ,F3,A4,46,46,47
      ,47
PJ 36Ø DATA 48,75,F4,AØ,Ø8,46,34
      ,Ø1
JD 37Ø DATA A2,Ø8,46,84,Ø5,5Ø,A8
      ,Ø1
GK 38Ø DATA 75,Ø5,88,ØØ,88,E8,Ø3
      ,88
MH 39Ø DATA ØØ,89,8E,CØ,89,4Ø,Ø1
      ,8F
NG 4ØØ DATA 6Ø,Ø4,8D,36,Ø8,41,8A
      ,DA
DF 41Ø DATA Ø3,EC,A8,Ø8,75,F8,EC
      ,A8
KO 42Ø DATA Ø8,74,F8,F3,A5,E8,Ø4
      ,E8
HC 43Ø DATA 8A,E8,AA,89,4Ø,Ø1,8A
      ,DA
EL 44Ø DATA Ø3,EC,A8,Ø8,75,F8,EC
      ,A8
FI 45Ø DATA Ø8,74,F8,F3,A5,58,CD
      ,1Ø
AF 46Ø DATA 89
FH 47Ø ' The following two value
      s are the
FL 48Ø ' time delay constant in
      the order
LJ 49Ø ' least sig. byte, most s
      ig. byte.
BO 5ØØ DATA Ø1,ØØ
BI 51Ø DATA E2,FE,59,E2,DF,Ø7,5E
      ,84
BI 52Ø DATA Ø6,82,FF,CD,21,3C,71
      ,74
MP 53Ø DATA Ø6,3C,51,74,Ø2,E8,CF
      ,88
FB 54Ø DATA E5,Ø7,8B,ØØ,Ø5,CD,1Ø
      ,C8
```

©

COMPUTE!
TOLL FREE
Subscription
Order Line
1-800-247-5470
In IA 1-800-532-1272

STATE-OF-THE-ART MAGNETIC MEDIA

5¼" DISKETTES

- With Hub Rings
- Write Protect Tebe
- Envelopee
- User ID Lebels
- In Fectory Seeled Poly Pecke of 10

(YOU GET EVERYTHING BUT THE BOX)
Prices ere per Diek

QTY	50	100	500	1000
SSDD	.59	.56	.52	.49
DSDD	64	.61	.57	.54

Library Case Holds 15 Diskettes Only $1 00!
plus 50¢ S&H
The 100 File Only $10 95! plus $2 00 S & H
100% ERROR FREE — LIFETIME WARRANTY
Min. order $25 00 Add 10% for less then 50 diskettes **Shipping end Hendling:** $4 00 per 100 diskettes Reduced shipping for lerger quantities C.O.D add $4 00 Cash or certified check

COD

Precision Data Products
P.O. Box 8367 Grand Rapids MI 49518
(616) 452-3457 • Michigan 1-800-632-2468
Outside Michigan 1-800-258-0028

Commodore 64 Screen Genie

James A Ledger

This thoughtfully designed utility helps you draw complete screens using character graphics. When you're finished, it writes a complete BASIC routine to recreate the graphics screen. The program runs on any Commodore 64 (or 128 in 64 mode) with either disk or tape.

"Commodore 64 Screen Genie" is both a screen editor and a program generator. With it, you can quickly and easily draw backgrounds for games, colorful title screens, or just pages of instructions. It offers a wealth of editing commands for designing a text or graphics character screen in normal, multicolor, or extended background mode. Then, almost instantly, it can write a BASIC routine to recreate that screen. This new routine is merged with whatever program is in memory. Since Screen Genie takes up no BASIC program space, it can be used with many other utilities such as the DOS Wedge, "TurboDisk," or "MetaBASIC."

Screen Genie is written entirely in machine language, so you'll need to enter it with the MLX machine language entry program found elsewhere in this issue. Follow the MLX instructions closely; here are the addresses you'll need for MLX:

Starting address: 0801
Ending address: 1D10

Built-In Help Screen

Screen Genie loads and runs like a normal BASIC program. Once you run it, however, the program breaks into several modules which move to various places, leaving the BASIC program space completely free (more on this process later).

The first thing you'll see is a help screen showing all of the Screen Genie commands. Fortunately, you don't have to memorize all the commands shown here. Since the help screen is always available, the only key sequence you need to remember is CTRL-H (hold down CTRL and press H). Selecting any command from the help screen returns you to the work screen and performs that command. Pressing any other key simply returns you to the work screen. Of course, all of the commands are also available directly from the work screen.

The help screen serves another purpose by indicating which modes and cursor functions are selected. For instance, if you select the Paint cursor function by pressing the f3 function key, a white arrow appears next to that option on the help screen.

Once you enter the work screen, almost all of the keys work as they normally do—text and graphics characters can be typed in whatever color you like. However, you may not type a quotation mark, insert a character by pressing SHIFT-INST/DEL, or break out of the program by pressing RUN/STOP-RESTORE. The delete key (DEL) is not disabled, but works in a slightly different way: It erases the character at the cursor position and moves the cursor one space left, but it doesn't drag any characters on the right with it.

Finally, to prevent the screen from scrolling, you are not allowed to type anything in the bottom right corner. Instead, this space is used to show the current color for the characters you're typing. That's a handy feature, since the cursor itself is no longer a blinking box. Instead, it's a blinking black and white underline.

Immediate Commands

Screen Genie's commands are divided into four groups: immediate commands, cursor functions, screen modes, and color selection. Here is an explanation of the immediate commands:

CTRL-H (Help). Display help screen.

CTRL-T (Top clear). Clear from the top of the screen to the current cursor position.

CTRL-B (Bottom clear). Clear from the bottom of the screen to the current cursor position.

CTRL-M (Move). Move a block of characters from one screen location to another. Before you can move a block, you must first define its upper-left and lower-right corners. Press CTRL-M, then place the cursor on the upper-left corner of the block you want to move, and press RETURN. Move the cursor to the lower right corner of the block, then press RETURN a second time. Now the block is defined. To move it elsewhere on the screen, move the cursor to the place where you want to put the upper-left corner of the new block, then press RETURN. The contents of the new area are replaced by the contents of the defined block (note that the original area is not disturbed). The Move command does not permit you to place the new block in any position that would overlap a screen border; all of the new block must fit inside the screen.

CTRL-Z (Memorize). Memorize the current screen by saving its contents in a memory buffer. A saved screen can be restored with CTRL-O.

CTRL-O (Oops). Swap the current screen with whatever is stored in the buffer. Pressing it again swaps it back. Besides restoring the screen after a manual save (CTRL-Z), this command can also undo any screen clear or move command.

CTRL-P (Program). Write a series of BASIC program lines to recreate the screen you've designed. These lines, beginning with the line number you choose, are merged with whatever BASIC program is in memory, if any. This feature performs a true merge, rather than simply tacking program lines onto the end of the current program. However, it does not replace any existing lines. If the merge operation would replace an existing program line, Screen Genie displays a message and gives you a chance to choose a new beginning line number.

The Program option also lets you add a line to set specific background and border colors. Likewise, if you're in extended background or multicolor mode when you choose this feature, you're given the option of adding lines that perform the setup for the current mode.

Finally, you have the option of adding a program line that waits for the user to press any key. This is useful for multiple pages of instructions, and so forth. Just be sure to include a prompt such as PRESS ANY KEY TO CONTINUE somewhere on the screen. If you choose this option while in extended background or multicolor mode, you may also add a line to turn the mode off after a key is pressed.

The default setting for all Program options is yes. Pressing any key other than Y or RETURN at the prompt selects no. If you have two or more sequential screens that use the same colors or mode, then you need only set these up on the first screen and turn the respective mode off on the last screen.

CTRL-X (Exit to BASIC). This lets you save, load, and edit BASIC programs as usual. Screen Genie is designed so that you can exit to BASIC and later reactivate the utility without disturbing a BASIC program in memory. To reactivate Screen Genie, just type GENIE and press RETURN.

Under ordinary circumstances, pressing RUN/STOP-RESTORE does not disable Screen Genie. If you disable it in some other way, type SYS 50800 and press RETURN to start it up again. When you reenter Screen Genie, the work screen contains whatever was on the screen when you left BASIC: Press CTRL-O immediately to recall what you were working on at the time of your last exit.

Since the GENIE command works in program mode as well as direct mode, you can edit a previously designed screen by inserting the word GENIE just after the last PRINT statement, and then running only that portion of the program that displays the screen. For example, if the routine that recreates your screen uses lines 500–525, then you could add GENIE to the end of line 525 (or the beginning of line 526) and type RUN 500. You'll need to give the new screen a different beginning line number, and then delete the old routine when you exit. (Don't forget to remove the GENIE command from the program when it's finished.)

Modes

In addition to ordinary text mode (what you see when you turn on the 64), Screen Genie lets you work in extended background color mode or multicolor mode, or replace the usual character set with a custom-defined character set of your own. Consult the *Commodore 64 User's Guide* for additional information on how to use these modes.

CTRL-K (Extended background). This mode permits each character to have any of four different background colors, but lets you use only the first 64 characters of the character set.

CTRL-C (Multicolor). Since the ordinary character set looks quite strange in multicolor mode, this mode will most likely require a custom character set. It cannot be used at the same time as extended background mode; selecting one mode turns the other off.

CTRL-U (User-defined characters). Selecting this mode causes the 64 to use a custom character set. Only the uppercase/graphics character set is available in this mode. Before choosing this option, you must store the character definitions in memory beginning at location 61440. Note that this configuration is only needed while you're editing the screen with Screen Genie. Once the screen design is done, and you have generated a BASIC routine to recreate the screen (see the Program option above), you can change your program to use whatever character set and memory locations you want.

Custom character mode demands a little more effort on your part. As in other cases, Screen Genie's Program option generates a complete routine with all of the necessary PEEKs, POKEs, and PRINTs needed to reproduce the screen. However, it's your job to put the custom character definitions in memory, decide on a location for the screen, and perform the extra POKEs needed to set everything up.

Cursor Functions

This group of options gives you additional control over the drawing cursor. They are selected by pressing one of the odd-numbered function keys. Any or all of these may be turned on at one time; however, if the Draw function (f1) is active, it takes precedence over the other three.

f1 (Draw with the cursor). This option lets you draw with any character. There are two ways to select the drawing character. You can either move the cursor to the desired character and press f1, or press f1 and type the character you want to use. To erase, press the space bar.

f3 (Paint with the cursor). Select a painting color just as you would normally change the cursor color in BASIC. Press CTRL or the Commodore key along with a number key from 1–8.

f5 (Change case with the cursor). This is very useful in extended background mode where a shifted character has a different background color.

f7 (Reverse characters with cursor). This option is also handy in extended background mode, where reversing a character gives it a different background color. When you reverse a space character in normal mode, it has whatever color happens to be stored in color memory—unless the color happens to be the same as the background color, in which case it is changed to the current text color in order to make it visible. You can guarantee the color of reversed spaces by turning on the Paint function at the same time.

Color Control

The even-numbered function keys provide you with complete color control as follows:

f2. Cycle the border color (memory location 53280).

f4. Cycle the normal background color (location 53281).

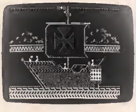
f6. The menu lets you cycle background color registers one, two, and three (these color registers are used only in extended background or multicolor mode).

f8. Cycle the color of every character that is the same color as the character under the cursor. If you continue to press f8, Screen Genie remembers which characters you started changing and cycles only those characters, rather than switching to new ones each time. As soon as you press any other key, however, these characters are forgotten.

Compatibility

Screen Genie is designed to coexist with other Commodore 64 utilities as peacefully as possible. To minimize memory conflicts, nearly all of its program code and workspace areas reside in the hidden RAM under the 64's BASIC ROM, Kernal ROM, and I/O address space. Even so, some not-so-hidden RAM had to be used. The memory locations from 50800–52223 ($C670–$CBFF) are used for links to the system, interrupt-driven routines, sprite shapes, and screen memory. This still leaves locations 49152–50799 ($C000–$C66F) free for programs such as "TurboDisk," and locations 52224–53247 ($CC00–$CFFF) free for programs such as the DOS Wedge. Programs which reside in the upper BASIC program area, such as "MetaBASIC," will not be affected at all.

If you want to use other utilities with this program, install them *before* you load Screen Genie. There is one minor quirk when using Screen Genie with MetaBASIC. Screen Genie wedges itself in through the BASIC error vector at 768–769, which is reset by some of the commands in MetaBASIC. No harm is done when this occurs—you'll just have to reenter Screen Genie with SYS 50800 rather than the more convenient GENIE command.



The "Screen Genie" screen editor/program generator makes it easy to create graphic screens like this and incorporate them in your own Commodore 64 BASIC programs.

Screen Genie For Commodore 64

Please refer to the "MLX" article in this issue before entering the following listing

```
Ø8Ø1:1B Ø8 ØA ØØ 9E 32 3Ø 37 B5
ØBØ9:34 3A 53 43 52 45 45 4E EØ
Ø811:2Ø 47 45 4E 49 45 ØØ ØØ EF
Ø819:ØØ A9 Ø4 85 ØE A9 ØD 85 23
Ø821:ØF A9 ØØ 85 1Ø A9 AØ 85 69
ØB29:11 A2 ØF AØ ØØ 88 B1 ØE E9
ØB31:91 1Ø CØ ØØ DØ F7 E6 ØF 69
Ø839:E6 11 CA 3Ø Ø6 DØ EE AØ 4F
Ø841:BB DØ EA AØ 8Ø B9 8B 1C 4F
ØB49:99 6F C6 BB DØ F7 AØ ØØ ØB
Ø851:98 99 ØØ C7 88 DØ FA A9 B7
Ø859:FF BD 15 C7 AØ 3F 99 4Ø 61
Ø861:C7 88 1Ø FA AØ 15 A9 CØ 96
Ø869:99 8Ø C7 88 88 88 1Ø F8 67
ØB71:A9 FF 8D 8Ø C7 BD 83 C7 53
Ø879:8D D2 C7 8D D5 C7 AØ ØF F4
Ø881:A9 Ø3 99 CØ C7 BB B8 BB 6Ø
Ø889:1Ø FB 78 A9 22 85 Ø1 A9 5C
Ø891:DØ AØ EØ A2 1Ø 2Ø EF AA Ø4
ØB99:A9 26 B5 Ø1 5B 2Ø 78 AB A8
Ø8A1:18 2Ø 92 A4 A9 Ø1 2Ø CF C3
Ø8A9:AA A9 ØØ AD 11 DØ 48 29 D9
Ø8B1:EF 8D 11 DØ A9 Ø9 A2 Ø9 ØC
Ø8B9:2Ø 9B A5 3B 2Ø 92 A4 68 F5
ØBC1:85 4B 2Ø C9 AA A9 17 A2 14
Ø8C9:ØA 2Ø 9B A5 A9 CB AØ BB 1F
Ø8D1:A2 Ø4 2Ø EF AA A9 DB AØ B5
Ø8D9:BC A2 Ø4 2Ø EF AA A9 E1 D2
Ø8E1:A2 Ø9 2Ø 9B A5 A9 ØØ 8D A4
Ø8E9:Ø1 Ø8 8D Ø2 Ø8 A9 Ø3 B5 CØ
ØBF1:2D 85 2F 85 31 A9 ØB 85 FD
Ø8F9:2E 85 3Ø 85 32 AD 11 DØ 1C
Ø9Ø1:Ø9 1Ø 8D 11 DØ 4C 39 A4 2D
Ø9Ø9:93 ØD ØD 2Ø 2Ø 1C 12 Ø2 63
Ø911:Ø8 2Ø 2A 2A 2A 2Ø 57 4F E6
Ø919:52 4B 2D 53 43 52 45 45 35
Ø921:4E 2Ø 2A 2A 2A Ø2 ØB 2Ø D3
Ø929:92 Ø2 Ø5 ØD Ø2 Ø4 2Ø 1E F4
Ø931:54 4F 2Ø 52 45 2D 45 4E 22
Ø939:54 45 52 2Ø 41 46 54 45 24
Ø941:52 2Ø 45 58 49 54 2Ø 54 E2
Ø949:4F 2Ø 42 41 53 49 43 ØD BA
Ø951:ØD Ø2 Ø4 2Ø 54 59 5Ø 45 DA
Ø959:2Ø 12 2Ø 47 45 4E 49 45 B3
Ø961:2Ø 92 2Ø 41 4E 44 2Ø 5Ø 54
Ø969:52 45 53 53 2Ø 52 45 54 BE
Ø971:55 52 4E ØD ØD Ø2 Ø4 2Ø F5
Ø979:98 28 2Ø 4F 52 2Ø 12 2Ø 32
Ø981:53 59 53 2Ø 35 3Ø 38 3Ø ØB
Ø9B9:3Ø 2Ø 92 2Ø 29 Ø2 Ø6 ØD 7A
Ø991:Ø2 Ø4 2Ø 81 5Ø 52 45 53 6B
Ø999:53 2Ø 12 53 48 49 46 54 1D
Ø9A1:2D 43 4C 52 92 2Ø 54 4F D6
Ø9A9:2Ø 43 4C 45 41 52 2Ø 53 61
Ø9B1:43 52 45 45 4E Ø2 Ø7 ØD 8C
Ø9B9:Ø2 Ø4 2Ø 1F 43 55 52 52 2A
Ø9C1:45 4E 54 2Ø 43 4F 4C 4F D5
Ø9C9:52 2Ø 49 4E 44 49 43 41 2A
Ø9D1:54 45 44 2Ø 48 45 52 45 2B
Ø9D9:2D 2D 2D 2D 2D 3E 13 ØØ CE
Ø9E1:98 13 1D 1D 43 4F 4D 5Ø BC
Ø9E9:55 54 45 21 1D 5Ø 55 42 8D
Ø9F1:4C 49 43 41 54 49 4F 4E AD
Ø9F9:53 1D 5Ø 52 45 53 45 4E 7C
ØAØ1:54 53 Ø2 Ø4 11 Ø2 Ø8 9D D2
ØAØ9:1D 1D 1D 1D 1D 1D 1D 1D 1D
ØA11:1D 1D 1D 1D 1D ØØ 93 9E 1F
ØA19:75 Ø2 26 63 69 62 2Ø 2Ø 98
ØA21:75 63 69 75 63 69 BØ 63 D2
ØA29:69 75 63 69 75 63 69 B2 11
ØA31:2Ø B2 2Ø 2Ø 75 63 69 75 89
ØA39:63 69 B2 2Ø B2 2Ø B2 75 A2
ØA41:63 69 2Ø 2Ø 62 62 2Ø 2Ø 64
ØA49:6A 63 69 62 2Ø 2Ø 62 63 68
ØA51:6B AB B3 2Ø AB B3 2Ø 62 4D
ØA59:6D 62 2Ø 2Ø 62 63 B2 AB 74
ØA61:B3 2Ø 62 6D 62 2Ø 62 AB 7E
ØA69:B3 2Ø 2Ø 2Ø 62 62 2Ø 2Ø 62
ØA71:6A 6Ø 6B 6A 63 6B B1 2Ø 33
ØA79:6D 6A 63 6B 6A 63 6B B1 6B
ØA81:2Ø B1 2Ø 2Ø 6A 63 6B 6A 3A
ØA89:63 6B B1 2Ø B1 2Ø B1 6A 3E
ØA91:63 6B 2Ø 2Ø 62 6A Ø2 26 1F
ØA99:63 6B 99 75 Ø2 12 6Ø 69 47
ØAA1:9C 75 Ø2 12 6Ø 69 99 62 Ø1
ØAA9:12 43 4F 4D 4D 41 4E 44 A6
ØAB1:53 3A 92 2Ø 2Ø 43 54 52 5B
ØAB9:4C 2Ø 26 2Ø 62 9C 62 12 1F
ØAC1:2Ø 43 55 52 53 4F 52 2Ø 23
ØAC9:46 55 4E 43 54 49 4F 4E Ø9
ØAD1:53 2Ø 92 62 99 62 Ø2 12 7C
ØAD9:2Ø 62 9C 62 Ø2 12 2Ø 62 4B
ØAE1:99 62 48 2D 48 45 4C 5Ø 77
ØAE9:2Ø 28 54 48 49 53 2Ø 4D 4C
ØAF1:45 4E 55 29 62 9C 62 46 ØA
ØAF9:31 2D 44 52 41 57 Ø2 ØA 15
ØBØ1:2Ø 5F 62 99 62 42 2D 43 9E
ØBØ9:4C 45 41 52 2Ø 54 4F 2Ø F4
ØB11:42 4F 54 54 4F 4D 2Ø 62 3E
ØB19:9C 62 46 33 2D 5Ø 41 49 B8
ØB21:4E 54 Ø2 Ø9 2Ø 5F 62 99 21
ØB29:62 54 2D 43 4C 45 41 52 AB
ØB31:2Ø 54 4F 2Ø 54 4F 5Ø Ø2 DA
ØB39:Ø4 2Ø 62 9C 62 46 35 2D 33
ØB41:43 48 41 4E 47 45 2Ø 43 EA
ØB49:41 53 45 2Ø 2Ø 2Ø 5F 62 22
ØB51:99 62 4D 2D 4D 4F 56 45 E2
ØB59:Ø2 ØC 2Ø 62 9C 62 46 37 CF
```

```
ØB61:2D 52 45 56 45 52 53 45 1Ø
ØB69:Ø2 Ø7 2Ø 5F 62 99 62 5A D4
ØB71:2D 4D 45 4D 4F 52 49 5A 9F
ØB79:45 2Ø 53 43 52 45 45 4E 59
Ø881:2Ø 62 9C 6A Ø2 12 6Ø 6B FE
ØB89:99 62 4F 2D 4F 4F 5Ø 53 6D
ØB91:21 2Ø 28 52 45 53 54 4F D9
Ø899:52 45 29 2Ø 62 9A 75 Ø2 88
Ø8A1:12 6Ø 69 99 62 5Ø 2D 43 91
Ø8A9:52 45 41 54 45 2Ø 5Ø 52 45
ØBB1:49 4E 54 2Ø 5Ø 47 4D 62 29
Ø889:9A 62 12 Ø2 Ø6 2Ø 43 4F 9E
ØBC1:4C 4F 52 Ø2 Ø7 2Ø 92 62 7C
ØBC9:99 62 58 2D 45 58 49 54 95
ØBD1:2Ø 54 4F 2Ø 42 41 53 49 FF
ØBD9:43 2Ø 2Ø 2Ø 62 9A 62 Ø2 E3
Ø8E1:12 2Ø 62 99 6A Ø2 12 6Ø CE
ØBE9:6B 9A 62 46 32 2D 42 4F 27
Ø8F1:52 44 45 52 Ø2 Ø9 2Ø 62 E6
Ø8F9:96 75 Ø2 12 6Ø 69 9A 62 5A
ØCØ1:46 34 2D 42 41 43 4B 47 Ø8
ØCØ9:52 4F 55 4E 44 Ø2 Ø5 2Ø Ø2
ØC11:62 96 62 12 2Ø 4D 4F 44 86
ØC19:45 53 3A 2Ø 92 2Ø 43 54 E1
ØC21:52 4C 2Ø 26 2Ø 2Ø 2Ø 62 FF
ØC29:9A 62 46 36 2D 42 41 43 88
ØC31:4B 47 4E 44 2Ø 52 45 47 EA
ØC39:2Ø 31 2D 33 62 96 62 Ø2 BA
ØC41:12 2Ø 62 9A 62 46 38 2D 2A
ØC49:43 48 41 52 53 2Ø 4F 46 62
ØC51:2Ø 53 41 4D 45 2Ø 2Ø 62 98
ØC59:96 62 48 2D 45 58 54 45 Ø8
ØC61:4E 44 45 44 2Ø 42 41 43 6E
ØC69:4B 47 44 5F 62 9A 62 2Ø D9
ØC71:2Ø 2Ø 43 4F 4C 4F 52 2Ø 63
ØC79:41 53 2Ø 43 48 41 52 2Ø 4B
ØC81:2Ø 62 96 62 43 2D 4D 55 F9
ØC89:4C 54 49 43 4F 4C 4F 52 D6
ØC91:Ø2 Ø5 2Ø 5F 62 9A 62 2Ø 48
ØC99:2Ø 2Ø 4F 56 45 52 2Ø 43 1Ø
ØCA1:55 52 53 4F 52 Ø2 Ø4 2Ø 1B
ØCA9:62 96 62 55 2D 55 53 45 E4
ØCB1:52 2Ø 44 45 46 49 4E 45 11
ØC89:44 2Ø 2Ø 2Ø 5F 62 9A 6A 26
ØCC1:Ø2 12 6Ø 6B 96 62 2Ø 2Ø CØ
ØCC9:43 48 41 52 2Ø 53 45 54 1Ø
ØCD1:2Ø 4Ø 2Ø 36 31 34 34 3Ø 64
ØCD9:62 98 2Ø 2Ø 5Ø 52 45 53 F8
ØCE1:53 2Ø 41 4E 59 2Ø 4B 45 DF
ØCE9:59 ØD 96 6A Ø2 12 6Ø 6B EF
ØCF1:98 2Ø 2Ø 46 4F 52 2Ø 57 22
ØCF9:4F 52 48 2D 53 43 52 45 1C
ØDØ1:45 4E ØØ AD 86 Ø2 8D 2C AF
ØDØ9:DØ 2Ø C3 A5 24 55 1Ø 19 16
ØD11:AØ 12 D9 37 AF FØ Ø5 88 82
ØD19:1Ø F8 3Ø ØD 98 ØA A8 89 48
ØD21:4A AF 48 C8 89 4A AF 48 8Ø
ØD29:6Ø C9 94 FØ D6 C9 14 DØ 5E
ØD31:ØA A9 95 A2 AC 2Ø 9B A5 5A
ØD39:4C ØØ AØ C9 22 FØ C4 C9 52
ØD41:93 DØ Ø6 18 2Ø 92 A4 A9 D9
ØD49:93 48 2Ø Ø8 AB 68 86 29 F8
ØD51:84 2A EØ 18 DØ ØC C9 ØD 2D
ØD59:FØ A9 C9 8D FØ A5 C9 11 2B
ØD61:FØ A1 24 ØC 5Ø 21 C9 9D D9
ØD69:FØ 1D C9 1D FØ 19 C9 91 5F
ØD71:FØ 15 C9 11 FØ 11 2Ø AC 4C
ØD79:AØ A6 29 A4 2A 18 2Ø FØ DF
ØD81:FF 2Ø C3 A1 4C 9A AØ 2Ø 64
ØD89:AC AØ 2Ø Ø8 AB EØ 18 DØ 88
ØD91:ØC CØ 27 DØ Ø8 A4 2A A6 A1
ØD99:29 18 2Ø FØ FF 2Ø 3Ø A6 E8
ØDA1:2Ø F5 AØ 2Ø BF AØ 2Ø DC FC
ØDA9:AØ 2Ø E8 AØ 4C ØØ AØ C9 8Ø
ØD81:ØD FØ Ø4 C9 8D DØ Ø8 E8 54
ØDB9:AØ ØØ 18 2Ø FØ FF 6Ø 4C BD
ØDC1:D2 FF 24 ØD 5Ø 18 AØ ØØ 8E
ØDC9:B1 ØE 49 8Ø 91 ØE 29 3F C7
ØDD1:C9 2Ø DØ ØA AD 21 DØ 29 5Ø
ØDD9:ØF CD 89 AF FØ Ø7 6Ø 24 A3
ØDE1:ØD 1Ø Ø7 AØ ØØ AD 86 Ø2 37
ØDE9:91 1Ø 6Ø 24 ØC 1Ø Ø8 AØ 7Ø
ØDF1:ØØ B1 ØE 49 4Ø 91 ØE 6Ø 93
ØDF9:24 ØC 5Ø ØA AØ ØØ A5 ØA 2E
ØEØ1:91 ØE A5 ØB 91 1Ø 6Ø 18 74
ØEØ9:2Ø 92 A4 2Ø 3Ø A6 A9 E7 C7
ØE11:85 1Ø A9 C8 85 11 2Ø 88 1F
ØE19:A6 4C ØØ AØ 18 2Ø 92 A4 BØ
ØE21:2Ø 3Ø A6 A5 ØE 85 1Ø A5 D4
ØE29:ØF 85 11 A9 ØØ 85 ØE A9 C6
ØE31:C8 85 ØF 2Ø 88 A6 4C ØØ 6E
ØE39:AØ EE 2Ø DØ 4C ØØ AØ EE Ø5
ØE41:21 DØ 4C ØØ AØ A5 ØD 49 AA
ØE49:8Ø 85 ØD 2Ø 3Ø A6 2Ø DC E3
ØE51:AØ 4C ØØ AØ A5 ØD 49 4Ø ØF
ØE59:85 ØD 2Ø 3Ø A6 2Ø BF AØ 58
ØE61:4C ØØ AØ 18 2Ø 92 A4 4C 1A
ØE69:ØØ AØ AD 11 DØ 49 4Ø 8D 2E
ØE71:11 DØ 29 4Ø FØ Ø8 AD 16 8C
ØE79:DØ 29 EF 8D 16 DØ 4C ØØ A8
ØE81:AØ AD 16 DØ 49 1Ø 8D 16 E4
ØE89:DØ 29 1Ø FØ Ø8 AD 11 DØ 53
ØE91:29 BF 8D 11 DØ 4C ØØ AØ 4D
ØE99:AD 18 DØ 49 Ø4 29 FC 8D 8D
ØEA1:18 DØ 4C ØØ AØ A5 ØC 49 84
ØEA9:8Ø 85 ØC 2Ø 3Ø A6 2Ø E8 3Ø
ØEB1:AØ 4C ØØ AØ A5 ØC 49 4Ø 6B
ØE89:85 ØC 2Ø 3Ø A6 2Ø C3 A1 81
ØEC1:2Ø F5 AØ 4C ØØ AØ AØ ØØ Ø8
ØEC9:81 ØE 85 ØA B1 1Ø 85 ØB 77
ØED1:6Ø 38 2Ø 92 A4 2Ø 3Ø A6 Ø6
ØED9:AØ ØØ B1 ØE 29 BF 2C 11 ØF
ØEE1:DØ 5Ø Ø2 29 3F C9 2Ø DØ 7F
ØEE9:Ø3 4C ØØ AØ 81 1Ø 29 ØF D3
ØEF1:85 4A A9 ØØ 85 ØE 85 1Ø 18
ØEF9:A9 D8 85 11 A9 FC 85 ØF 3E
ØFØ1:AØ ØØ A2 Ø4 2Ø 81 AA B1 12
ØFØ9:ØE 2Ø 87 AA 29 ØF C5 4A 2D
ØF11:DØ Ø9 81 1Ø 18 69 Ø1 29 A2
ØF19:ØF 91 1Ø C8 DØ E6 E6 11 82
ØF21:E6 ØF CA DØ DF 2Ø C3 A5 89
ØF29:C9 8C FØ C6 4C Ø9 AØ 38 D9
ØF31:2Ø 92 A4 2Ø CD AA A9 A9 BØ
ØF39:A2 AE 2Ø 9B A5 A9 31 85 CD
ØF41:1Ø 8D F9 AE A9 D6 A2 AE 91
ØF49:2Ø 9B A5 E6 1Ø A5 1Ø C9 82
ØF51:34 DØ EE A9 ØD A2 AF 2Ø A8
ØF59:9B A5 A9 1C 8D 15 DØ AØ A8
ØF61:Ø2 B9 43 ØØ 99 29 DØ 88 F2
ØF69:1Ø F7 2Ø F3 A9 C9 2Ø DØ 56
ØF71:ØC A9 ØØ 8D 15 DØ 38 2Ø 55
ØF79:ED A4 4C ØØ AØ C9 31 9Ø 6Ø
ØF81:E9 C9 34 BØ E5 38 E9 3Ø AC
ØF89:A8 B6 42 E8 8A 99 42 ØØ 8F
ØF91:4C 5C A2 18 2Ø 92 A4 2Ø 77
ØF99:AC A3 A9 18 85 3B A9 27 41
ØFA1:85 3C A9 1E 8D F8 C8 2Ø BØ
ØFA9:B6 A3 86 29 86 39 84 2A B8
ØFB1:84 3A A5 ØE 85 4C A5 ØF ED
ØF89:85 4D A5 1Ø 85 4E A5 11 65
ØFC1:85 4F AD ØØ DØ 8D ØC DØ D1
ØFC9:AD Ø1 DØ 8D ØD DØ AD 1Ø Ø9
ØFD1:DØ 29 BF 8D 1Ø DØ 29 Ø1 8A
ØFD9:4A 6A 6A ØD 1Ø DØ 8D 1Ø C4
ØFE1:DØ A9 Ø2 8D 2D DØ 8D 2E E1
ØFE9:DØ A9 6Ø 8D 15 DØ A9 1F 1E
ØFF1:8D F8 CB 2Ø B6 A3 8A 38 22
ØFF9:E5 29 85 3D 98 38 E5 2A 75
1ØØ1:85 3E A9 18 38 E5 3D 85 83
1ØØ9:3B A9 27 38 E5 3E 85 3C Ø9
1Ø11:AD ØØ DØ 8D ØE DØ AD Ø1 ØB
1Ø19:DØ 8D ØF DØ AD 1Ø DØ 29 6C
1Ø21:7F 8D 1Ø DØ 29 Ø1 4A 6A BF
1Ø29:ØD 1Ø DØ 8D 1Ø DØ A9 EØ BE
1Ø31:8D 15 DØ A6 29 A4 2A 18 2A
1Ø39:2Ø FØ FF A9 1E 8D F8 CB 25
1Ø41:2Ø AC A3 2Ø B6 A3 A5 ØE BØ
1Ø49:85 5Ø A5 ØF 85 51 A5 1Ø B2
1Ø51:85 52 A5 11 85 53 A5 4C 9F
1Ø59:85 ØE A5 4D 85 ØF A9 ØØ Ø5
1Ø61:85 1Ø A9 F8 85 11 2Ø E7 A5
1Ø69:A3 A5 4E 85 ØE A5 4F B5 12
1Ø71:ØF A9 ØØ 85 1Ø A9 FC 85 82
1Ø79:11 2Ø E7 A3 A9 ØØ 85 ØE C7
1Ø81:A9 F8 85 ØF A5 5Ø 85 1Ø DF
1Ø89:A5 51 85 11 2Ø E7 A3 A9 24
1Ø91:ØØ 85 ØE A9 FC 85 ØF A5 31
1Ø99:52 85 1Ø A5 53 85 11 2Ø 93
1ØA1:E7 A3 A9 2Ø 8D 15 DØ A9 E1
1ØA9:1C 8D F8 C8 4C ØØ AØ A9 64
1ØB1:ØØ 85 39 85 3A 6Ø 2Ø D2 19
1ØB9:FF 2Ø C3 A5 48 2Ø Ø8 AB 33
1ØC1:68 C9 91 DØ Ø6 E4 39 DØ CE
1ØC9:ED FØ EE C9 11 DØ Ø6 E4 54
1ØD1:3B DØ E3 FØ E4 C9 9D DØ A9
1ØD9:Ø6 C4 3A DØ D9 FØ DA C9 94
1ØE1:1D DØ Ø6 C4 3C DØ CF FØ 87
1ØE9:DØ 6Ø A6 3D 2Ø 81 AA A4 34
1ØF1:3E B1 ØE 91 1Ø A9 2Ø 91 71
1ØF9:ØE 88 1Ø F5 2Ø 87 AA A5 BE
11Ø1:ØE 18 69 28 85 ØE 9Ø Ø2 67
11Ø9:E6 ØF A5 1Ø 18 69 28 85 54
1111:1Ø 9Ø Ø2 E6 11 CA 1Ø D4 B6
1119:6Ø 18 2Ø ED A4 4C ØØ AØ 4B
1121:2Ø 78 AB 38 2Ø 92 A4 2Ø 1F
1129:C9 AA A9 B8 AØ C8 A2 Ø4 ØD
1131:2Ø EF AA A9 BC AØ D8 A2 ØC
1139:Ø4 2Ø EF AA A5 ØC 29 CØ 7E
1141:ØA AA 2A 8D B6 D9 8A ØA 6D
1149:2A 8D 66 D9 A5 ØD 29 CØ C2
1151:ØA AA 2A 8D 8E D9 8A ØA 3C
1159:2A 8D DE D9 A5 48 29 4Ø 4E
1161:ØA ØA 2A 8D 32 DB A5 47 BC
1169:29 1Ø FØ Ø2 A9 Ø1 8D 5A 29
1171:D8 A5 46 29 Ø4 4A 4A 8D B1
1179:82 DB A9 ØØ 85 C6 2Ø DC 6D
1181:A5 24 55 3Ø Ø7 38 2Ø ED 74
1189:A4 4C ØØ AØ 48 38 2Ø ED 6C
1191:A4 68 4C Ø9 AØ 8Ø ØD A9 C5
1199:88 85 ØE A9 AF 85 ØF A9 19
11A1:8Ø 48 DØ ØB A9 41 85 ØE 64
11A9:A9 ØØ 85 ØF A9 F8 48 AØ A4
11B1:Ø7 AD 11 DØ 91 ØE 88 AD 75
11B9:16 DØ 91 ØE 88 AD 18 DØ 2A
11C1:91 ØE 88 B9 2Ø DØ 29 ØF 82
11C9:91 ØE 88 1Ø F6 AD 86 Ø2 C7
11D1:85 49 A5 C7 85 48 2Ø Ø8 DB
11D9:A8 86 29 84 2A 68 A8 A9 CE
11E1:C8 A2 Ø4 2Ø EF AA A4 11 18
11E9:A9 D8 A2 Ø4 2Ø EF AA 6Ø 22
11F1:Ø8 BØ 11 38 2Ø 92 A4 A9 28
11F9:88 85 ØE A9 AF 85 ØF A9 79
12Ø1:BØ 48 DØ 1C A9 41 85 ØE D6
12Ø9:A9 ØØ 85 ØF A9 F8 48 A5 Ø8
1211:48 85 C7 A5 49 8D 86 Ø2 1F
1219:A6 29 A4 2A 18 2Ø FØ FF 35
1221:AØ Ø7 B1 ØE 8D 11 DØ 88 49
1229:81 ØE 8D 16 DØ 88 81 ØE D6
1231:8D 18 DØ 88 B1 ØE 99 2Ø DD
1239:DØ 88 1Ø F8 2Ø 81 AA AØ 76
1241:C8 A2 Ø4 68 2Ø EF AA A5 35
1249:ØF AØ D8 A2 Ø4 2Ø EF AA 8D
1251:2Ø 87 AA 28 8Ø 1A 2Ø 81 EE
1259:AA A9 F8 AØ BØ A2 Ø8 2Ø A6
1261:EF AA 2Ø 87 AA AØ Ø7 B9 44
1269:41 ØØ 99 88 AF 88 1Ø F7 A1
1271:A9 2Ø 8D 15 DØ 6Ø AØ ØØ BE
1279:A6 1Ø DØ Ø2 C6 11 C6 1Ø 47
1281:A6 ØE DØ Ø2 C6 ØF CA 86 45
1289:ØE 2Ø 31 AC B1 ØE 2Ø 39 EC
1291:AC 91 1Ø E4 5F DØ E1 A6 69
1299:ØF E4 6Ø DØ D8 6Ø 85 ØE 11
12A1:86 ØF AØ ØØ B1 ØE FØ ØE 96
12A9:C9 Ø2 FØ ØB 2Ø D2 FF C8 17
12B1:DØ F2 E6 ØF DØ EE 6Ø C8 94
12B9:81 ØE AA C8 B1 ØE 2Ø D2 F4
12C1:FF CA DØ FA FØ E9 2Ø 3Ø Ø2
12C9:A6 A9 Ø1 ØD 15 DØ 8D 15 B8
12D1:DØ AD 27 DØ 49 Ø1 8D 27 4C
12D9:DØ A9 14 85 13 85 12 2Ø 9E
12E1:F3 A9 AA A5 91 49 FB DØ 94
12E9:Ø6 A9 8Ø 85 55 3Ø ØC EØ 48
12F1:85 9Ø Ø4 EØ 8D 9Ø F2 A9 C9
```

```
12F9:00 85 55 A0 00 84 12 A9 14
1301:FE 2D 15 D0 8D 15 D0 8A 8E
1309:60 18 20 92 A4 A2 22 BD 44
1311:90 AF 95 08 CA 10 F8 A2 C9
1319:22 8D 83 AF 95 39 CA 10 68
1321:F8 78 A2 D6 A0 AF 18 20 87
1329:8D FF A9 04 8D 88 02 58 76
1331:4C 76 C6 A9 00 85 0F 85 48
1339:10 38 20 F0 FF 98 C9 28 A6
1341:90 03 E9 28 A8 0A 0A 0A 88
1349:26 10 69 18 8D 00 D0 90 D3
1351:02 E6 10 AD 10 D0 29 FE 24
1359:05 10 8D 10 D0 8A 0A 0A 87
1361:0A 85 10 69 32 8D 01 D0 21
1369:A5 10 0A 26 0F 0A 26 0F 06
1371:18 65 10 90 02 E6 0F 84 56
1379:10 18 65 10 85 0E 85 10 DA
1381:A9 C8 65 0F 85 0F 69 10 97
1389:85 11 60 A0 00 A9 20 91 45
1391:0E A6 0E E4 10 D0 06 A6 EE
1399:0F E4 11 F0 08 E6 0E D0 7A
13A1:EE E6 0F D0 EA 60 A9 20 34
13A9:8D E7 CB 38 20 92 A4 20 42
1381:CD AA A9 93 20 D2 FF A9 CD
1389:00 85 14 85 15 85 40 A9 05
13C1:FF A2 AC 20 98 A5 A9 15 04
13C9:A2 AF 20 9B A5 20 C3 A5 C5
13D1:4C E8 A6 38 20 ED A4 4C FE
13D9:00 A0 A9 86 A2 AC 20 98 6C
13E1:A5 4C B4 A6 20 C3 A5 C9 14
13E9:0D F0 43 C9 20 F0 E4 C9 30
13F1:30 90 E7 C9 3A 80 E3 20 6A
13F9:D2 FF 29 0F A8 A5 14 A6 4A
1401:15 06 14 26 15 80 D3 06 33
1409:14 26 15 80 CD 65 14 85 24
1411:14 8A 65 15 80 C4 85 15 9C
1419:06 14 26 15 80 8C 98 65 6E
1421:14 85 14 A5 15 69 00 80 90
1429:81 85 15 4C E1 A6 A5 14 F8
1431:85 1C A5 15 C9 FA 90 07 8B
1439:A9 DC A2 AC 4C D8 A6 85 31
1441:1D A9 00 85 1A A9 D0 85 59
1449:18 A5 47 29 10 F0 1D 20 82
1451:15 A8 20 53 A8 20 6D A8 8C
1459:D0 30 A9 00 8D 87 AD 20 F1
1461:0E AA A9 D6 A2 AD 20 41 2B
1469:AA 4C 87 A7 24 48 50 1A 62
1471:20 15 AB 20 53 A8 20 6D 5D
1479:A8 D0 0F A9 3A 8D 87 AD 4D
1481:20 0E AA A9 FD A2 AD 20 23
1489:41 AA 20 0E AA A5 41 20 70
1491:58 AA 8D 7E AD 8E 7D AD F8
1499:A5 42 20 5B AA 8D 88 AD 29
14A1:8E 87 AD A5 46 29 02 F0 CE
14A9:04 A9 FF D0 02 A9 32 8D F3
1481:BF AD 20 53 AB A9 72 A2 D1
1489:AE 20 9B A5 20 F9 A9 D0 1C
14C1:07 A9 76 A2 AD 4C C9 A7 AA
14C9:A9 8A A2 AD 20 41 AA 20 14
14D1:6C AA 24 48 70 26 20 E0 21
14D9:A9 A0 27 20 CC A9 C9 60 E6
14E1:F0 0F C9 A0 F0 04 C9 E0 95
14E9:D0 0B 20 D5 A9 C5 42 D0 58
14F1:04 A9 20 91 16 88 10 E3 7A
14F9:C6 26 D0 DD 20 E0 A9 A9 88
1501:10 85 08 A0 00 20 CC A9 63
1509:85 09 4C 0E A8 A9 92 85 39
1511:09 A2 00 86 54 CA 86 24 CF
1519:20 0E AA 20 8F AA A0 27 BD
1521:20 CC A9 C9 20 D0 12 88 51
1529:10 F6 A9 11 20 2E AA 20 8E
1531:9A AA C6 26 D0 E8 4C D5 27
1539:A8 84 25 E6 24 A4 24 20 08
1541:CC A9 AA 30 0E 24 09 30 D7
1549:15 A9 92 85 09 20 2E AA E2
1551:4C 5B A8 24 09 10 07 A9 10
1559:12 85 09 20 2E AA 8A 24 66
1561:48 50 02 29 3F C9 20 F0 E8
1569:12 A4 24 20 D5 A9 C5 08 35
1571:F0 09 85 08 A8 B9 70 AF 44
1579:20 2E AA 8A 29 7F C9 20 38
```

```
1581:90 08 C9 40 90 0A C9 60 D3
1589:90 03 18 69 20 18 69 20 AA
1591:20 2E AA A5 24 C5 25 F0 7A
1599:16 A5 54 C9 47 90 9C 20 35
15A1:AA AA 20 0E AA 20 8F AA 50
15A9:A9 00 85 54 4C 38 A8 20 53
1581:9A AA C6 26 F0 1C A5 24 76
15B9:C9 27 D0 06 20 AA AA 4C 5A
15C1:0E A8 A9 22 20 2E AA A9 2D
15C9:00 20 2E AA 20 6C AA 4C C0
15D1:0A A8 20 AA AA 20 6C AA 33
15D9:A5 20 85 1A A5 21 85 18 09
15E1:A9 22 A2 AD 20 98 A5 20 73
15E9:F9 A9 D0 46 A5 27 85 14 E2
15F1:A5 28 85 15 20 0E AA 20 A9
15F9:6C AA A9 96 A2 AD 20 41 F0
1601:AA A5 47 29 10 F0 15 20 F5
1609:5B AB D0 26 20 0E AA 20 F8
1611:6C AA A9 E9 A2 AD 20 41 3F
1619:AA 4C 2F A9 24 48 50 12 23
1621:20 58 AB D0 0D 20 0E AA 66
1629:20 6C AA A9 C2 A2 AD 20 8C
1631:41 AA A5 3F F0 1F A9 36 DE
1639:A2 AD 20 9B A5 A6 1C A5 85
1641:1D 20 41 AC A9 53 A2 AD 84
1649:20 98 A5 A6 27 A5 28 20 CB
1651:41 AC 4C 84 A6 A5 27 D0 09
1659:02 C6 28 C6 27 A5 28 C9 93
1661:FA 90 06 20 85 AA 4C 35 18
1669:A7 A5 1A 38 E9 00 85 10 04
1671:A5 18 E9 D0 85 11 A5 2D 6A
1679:85 0E 18 65 10 85 10 A8 A4
1681:A5 2E 85 0F 65 11 85 11 39
1689:C5 38 90 1A D0 04 C4 37 81
1691:90 14 A9 57 A2 AD 20 98 5D
1699:A5 A9 15 A2 AF 20 98 A5 AA
16A1:20 F3 A9 4C D0 A6 A2 05 40
16A9:95 2D CA 94 2D CA 10 F8 3C
16B1:20 73 A5 A5 1A 85 0E A5 82
1689:1B 85 0F A9 00 85 5F A9 CF
16C1:D0 85 60 20 73 A5 20 7C 84
16C9:C6 20 B5 AA 4C 96 A9 20 F2
16D1:81 AA 81 16 20 87 AA 60 D5
16D9:20 81 AA 81 18 20 87 AA E1
16E1:29 0F 60 A9 19 85 26 A9 E1
16E9:00 85 16 85 18 A9 F8 85 71
16F1:17 A9 FC 85 19 60 20 E4 78
16F9:FF F0 F8 60 20 E4 FF F0 6D
1701:FB C9 59 F0 08 C9 0D F0 64
1709:07 08 A9 4E 20 D2 FF 28 4B
1711:60 A9 FF 20 2E AA 20 2E 66
1719:AA A5 14 20 2E AA A5 15 07
1721:20 2E AA 20 85 C6 90 02 AC
1729:E6 40 E6 14 D0 02 E6 15 6A
1731:60 20 31 AC A0 00 91 1A CA
1739:20 39 AC E6 1A D0 02 E6 C8
1741:18 E6 54 60 85 10 86 11 D1
1749:A0 00 81 10 C9 FF F0 03 32
1751:20 2E AA E6 10 D0 02 E6 8D
1759:11 C9 00 D0 E8 60 A2 FF B5
1761:29 0F C9 0A 90 06 A2 31 D4
1769:18 69 26 60 69 30 60 A5 3B
1771:1A 85 20 A5 18 85 21 A5 43
1779:14 85 27 A5 15 85 28 A5 07
1781:40 85 3F 60 78 A9 25 85 59
1789:01 60 48 A9 26 85 01 68 A5
1791:58 60 A9 99 20 2E AA A9 8B
1799:22 20 2E AA 60 A9 28 18 63
17A1:65 16 85 16 85 18 90 04 CB
17A9:E6 17 E6 19 60 A9 93 A2 F2
17B1:AD 20 41 AA 20 6C AA 60 F9
17B9:A9 99 A2 AC 20 9B A5 A5 A2
17C1:28 A6 27 20 41 AC A9 0D B1
17C9:20 D2 FF 60 A9 00 F0 02 F3
17D1:A9 0F 8D 21 D0 8D 20 D0 2A
17D9:A9 28 8D 18 D0 A9 C8 8D 66
17E1:16 D0 AD 11 D0 29 10 09 6A
17E9:0B 8D 11 D0 A9 00 8D 15 AD
17F1:D0 60 85 0F 84 11 A0 00 EB
17F9:84 0E 84 10 B1 0E 91 10 78
1801:88 D0 F9 E6 0F E6 11 CA 58
```

```
1809:D0 F2 60 38 20 F0 FF C0 73
1811:28 90 04 98 E9 28 A8 60 25
1819:20 53 AB A9 55 A2 AE 20 F0
1821:98 A5 20 F9 A9 D0 2E 20 39
1829:0E AA 20 6C AA A5 43 20 68
1831:5B AA 8D AC AD 8E AB AD E2
1839:A5 44 20 5B AA 8D B6 AD AD
1841:8E B5 AD A5 45 20 58 AA 42
1849:8D C0 AD 8E 8F AD A9 A4 88
1851:A2 AD 20 41 AA 60 A9 10 90
1859:A2 AE 20 98 A5 60 A9 91 D7
1861:A2 AE 20 98 A5 A9 26 A2 0F
1869:AE 20 9B A5 20 F9 A9 60 63
1871:A9 26 A2 AE 20 98 A5 20 1A
1879:F9 A9 60 A2 22 85 08 9D DC
1881:90 AF CA 10 F8 A2 22 85 8C
1889:39 9D 83 AF CA 10 F8 A2 5A
1891:D6 A0 AF 38 20 8D FF 78 7E
1899:20 8A FF A9 EF 8D 18 03 FF
18A1:A9 C6 8D 19 03 A9 9A 8D 1D
18A9:14 03 A9 C6 8D 15 03 58 65
1881:A9 00 8D 27 D0 85 0D 85 17
18B9:12 85 0C 8D 17 D0 AD 01 07
18C1:03 C9 C6 F0 13 8D C2 C6 E8
18C9:AD 00 03 8D C1 C6 A9 8C 43
18D1:8D 00 03 A9 C6 8D 01 03 35
18D9:A9 04 A0 C8 A2 04 20 EF D5
18E1:AA AD 02 DD 09 03 8D 02 62
18E9:DD A9 00 8D 00 DD AD 18 37
18F1:D0 29 02 09 28 8D 18 D0 1E
18F9:A9 C8 8D 88 02 A9 1C 8D E7
1901:F8 C8 A0 04 A9 1D 99 F9 E5
1909:CB 88 D0 FA A9 1E 8D FE EC
1911:CB A9 1F 8D FF CB A0 07 C7
1919:89 80 AF 99 04 D0 88 10 5C
1921:F7 A9 3C 8D 10 D0 A9 1C 4D
1929:8D 1D D0 8D 1B D0 A9 13 DE
1931:20 D2 FF 60 78 48 A9 24 8A
1939:85 01 68 60 48 A9 26 85 3C
1941:01 68 58 60 86 56 85 57 0F
1949:A2 03 A9 00 95 58 CA 10 76
1951:F8 A0 0F 06 56 26 57 78 5E
1959:F8 A5 58 65 58 85 58 A5 02
1961:59 65 59 85 59 A5 5A 65 98
1969:5A 85 5A D8 58 88 10 E3 E8
1971:A2 02 85 58 48 4A 4A 4A F8
1979:4A 20 8A AC 68 29 0F 20 1B
1981:8A AC CA 10 ED A5 58 D0 0C
1989:03 20 8E AC 60 C5 5B F0 A3
1991:F8 09 30 85 5B 4C D2 FF 14
1999:20 9D 9D 00 93 1F 02 0A 1E
19A1:0D 4C 41 53 54 20 4C 49 CF
19A9:4E 45 20 4E 55 4D 42 45 E6
19B1:52 20 55 53 45 44 3A 20 C4
19B9:00 93 12 1C 2A 2A 2A 20 43
19C1:49 4E 56 41 4C 49 44 20 38
19C9:45 4E 54 52 59 20 2A 2A A8
19D1:2A 92 20 54 52 59 20 41 80
19D9:47 41 49 4E 2E 0D 00 0D C0
19E1:1C 4C 49 4E 45 20 4E 55 DF
19E9:4D 42 45 52 20 45 58 43 2B
19F1:45 45 44 53 20 36 33 39 4F
19F9:39 39 20 4C 49 4D 49 54 46
1A01:0D 00 0D 0D 97 45 4E 54 F0
1A09:45 52 20 42 45 47 49 4E C4
1A11:4E 49 4E 47 20 4C 49 4E 10
1A19:45 20 4E 55 4D 42 45 52 67
1A21:20 4F 52 0D 00 0D 0D 41 E3
1A29:44 44 20 57 41 49 54 20 02
1A31:4C 4F 4F 50 3F 20 59 9D 19
1A39:00 93 1C 02 0A 0D 45 58 5D
1A41:49 53 54 49 4E 47 20 4C 2A
1A49:49 4E 45 53 20 42 45 54 7C
1A51:57 45 45 4E 20 00 20 26 77
1A59:20 00 93 02 0A 0D 1C 12 FE
1A61:2A 2A 2A 20 4F 55 54 20 15
1A69:4F 46 20 4D 45 4D 4F 52 FF
1A71:59 20 2A 2A 2A 92 0D 0D 05
1A79:00 97 35 33 32 38 30 2C 6C
1A81:30 30 3A 97 35 33 32 38 AD
1A89:31 2C 30 30 3A 99 C7 28 5A
1A91:31 34 32 29 22 93 22 38 23
```

```
1A99:00 97 31 39 38 2C 30 3A 7A
1AA1:92 31 39 38 2C 31 00 97 D3
1AA9:35 33 32 38 32 2C 30 30 E1
1AB1:3A 97 35 33 32 38 33 2C C7
1AB9:30 30 3A 97 35 33 32 38 E5
1AC1:34 2C 30 30 00 97 35 33 20
1AC9:32 36 35 2C C2 28 35 33 62
1AD1:32 36 35 29 AF 31 39 31 CB
1AD9:00 97 35 33 32 37 30 2C C8
1AE1:C2 28 35 33 32 37 30 29 53
1AE9:B0 31 36 00 97 35 33 32 B3
1AF1:37 30 2C C2 28 35 33 32 2E
1AF9:37 30 29 AF 32 33 39 00 C6
1B01:97 35 33 32 36 35 2C C2 7B
1B09:28 35 33 32 36 35 29 B0 B3
1B11:36 34 00 0D 0D 41 44 44 7A
1B19:20 4C 49 4E 45 20 54 4F 23
1B21:20 53 45 54 20 55 50 0D 2E
1B29:00 45 58 54 45 4E 44 45 32
1B31:44 20 42 41 43 4B 47 52 16
1B39:4F 55 4E 44 20 4D 4F 44 93
1B41:45 20 4F 52 0D 4D 55 4C C5
1B49:54 49 43 4F 4C 52 20 4D 92
1B51:4F 44 45 3F 20 59 9D 00 7E
1B59:42 41 43 4B 47 52 4F 55 95
1B61:4E 44 20 52 45 47 49 53 26
1B69:54 45 52 53 20 31 2D 33 ED
1B71:3F 20 59 9D 00 42 41 43 23
1B79:4B 47 52 4F 55 4E 44 20 F2
1B81:26 20 42 4F 52 44 45 52 90
1B89:20 43 4F 4C 4F 52 53 3F F8
1B91:20 59 9D 00 0D 0D 41 44 45
1B99:44 20 4C 49 4E 45 20 54 34
1BA1:4F 20 54 55 52 4E 20 4F C2
1BA9:46 46 0D 00 93 0D 1C 12 51
1BB1:02 05 20 50 52 45 53 53 D4
1BB9:20 4B 45 59 53 20 31 2D BB
1BC1:33 20 54 4F 20 43 59 43 1D
1BC9:4C 45 20 43 4F 4C 4F 52 4C
1BD1:53 02 05 20 92 02 03 0D 84
1BD9:00 0D 97 02 1D 20 6F 02 B0
1BE1:04 B7 70 0D 02 05 20 42 BD
1BE9:41 43 4B 47 52 4F 55 4E 38
1BF1:44 20 52 45 47 49 53 54 4B
1BF9:45 52 20 20 30 20 20 A5 55
1C01:02 04 20 A7 0D 02 1D 20 84
1C09:6C 02 04 AF BA 0D 0D 00 97
1C11:0D 0D 0D 12 1F 02 06 20 03
1C19:50 52 45 53 53 20 27 53 A8
1C21:50 41 43 45 27 20 46 4F 24
1C29:52 20 57 4F 52 4B 2D 53 DF
1C31:43 52 45 45 4E 02 05 20 41
1C39:92 00 89 8A 8B 8C 85 86 B4
1C41:87 88 02 14 0D 0F 08 10 A5
1C49:18 1A 0B 03 15 A1 35 A1 E0
1C51:3B A2 2B A1 CD A1 B0 A1 47
1C59:41 A1 A1 A1 50 A1 03 A1 99
1C61:18 A2 8F A4 15 A4 1F A6 AA
1C69:A2 A6 05 A1 5F A1 66 A1 47
1C71:7D A1 94 90 05 1C 9F 9C E0
1C79:1E 1F 9E 81 95 96 97 98 43
1C81:99 9A 9B 00 63 00 8B 00 D2
1C89:83 50 F2 20 8E C6 4C 1D 55
1C91:A4 20 93 C6 4C 69 FE 20 29
1C99:93 C6 20 33 A5 4C 8E C6 C6
1CA1:20 93 C6 20 13 A6 4C 8E 04
1CA9:C6 A9 26 B5 01 60 48 A9 90
1CB1:27 B5 01 68 60 A5 12 F0 34
1CB9:1B C6 13 D0 17 A9 14 85 AD
1CC1:13 AD 27 D0 49 01 8D 27 71
1CC9:D0 AD 2D D0 49 01 BD 2D 1F
1CD1:D0 8D 2E D0 4C 31 EA E0 86
1CD9:0B F0 03 4C 8B E3 A5 7A AA
1CE1:8D DB C6 A5 7B 8D DC C6 9D
1CE9:A0 04 AD DB C6 D0 03 CE 35
1CF1:DC C6 CE DB C6 AD 01 08 D8
1CF9:D9 EA C6 D0 DE 88 10 EA E3
1D01:68 68 4C 70 C6 47 45 4E 46
1D09:49 45 40 AA 00 00 00 00 EB
                                ©
```

Screen Saver 64

Stephen E. Masters

Here are two fast, useful routines for storing and retrieving high-resolution graphics screens with a disk drive. They work with the Commodore 64 or Commodore 128 in 64 mode.

Taking advantage of the Commodore 64's high-resolution graphics can be a time-consuming process at best. Even with extra commands such as those found in Simons' BASIC, it may take many minutes or even hours to plot a detailed screen. Utilities for dumping a high-resolution screen to your dot-matrix printer are readily available. At times, however, you may wish to save your graphics screen in a disk file so you can display it later without rerunning the program that created it. And if you own an Okimate 10 or similar color printer, the ability to save multicolor graphics screens is particularly useful.

"Screen Saver 64" provides two machine language (ML) routines that let you quickly save and retrieve hi-res graphics screens—both standard and multicolor—from disk. Though they're written in ML, you can use them without knowing the ins and outs of ML yourself. And we've included two demonstration programs that show exactly how to use the ML routines for real applications.

To get started, type in and save Programs 1 and 2. Program 1 puts the screen save routine into memory, and Program 2 creates the screen retrieval routine. Since both ML routines go into the same memory area, they must be used separately. If you have an ML monitor and wish to examine the routines, note that each is broken into two parts, located from memory locations 679–738 and 828–1023 (decimal).

Saving A Graphics Screen

Here are the steps for saving a graphics screen with Screen Saver 64:

1. Run Program 1 to place the ML screen save routine in memory.

2. Create your hi-res or multicolor screen as usual. If you don't know how to do this, the *Commodore 64*

Programmer's Reference Guide and many other books explain the required steps. Program 3 (see below) contains a simple demonstration.

3. Execute a statement like OPEN 2,8,2,"*filename*,P,W" to open a disk file for writing (replace *filename* with the name of your own file). You must open the file as a PRG (program format) file using the ,P suffix as shown above. The ,W suffix indicates that you're opening the file for a write operation, and the first numeral 2 sets the logical file number (2 in this case) for that file.

4. Execute SYS 1007 to activate the ML save routine. *This must be done while you are in hi-res or multicolor mode.* The ML routine finds the currently defined graphics screen and associated color memory, and stores their contents in the disk file.

5. Execute a statement like CLOSE 2 to close the file. It is *very important* that you end the procedure by CLOSEing the file, specifying the same logical file number (2 in this case) which was used to open it. If you omit this vital step, you may end up with a poison (unclosed) file on the disk that could damage other files or render the whole disk unreadable.

Retrieving A Graphics Screen

Once you have saved the screen to disk, it's easy to retrieve. Here are the steps to follow for bringing a graphics screen back into memory:

1. Run Program 2 to put the ML retrieval routine in memory.

2. Perform the steps needed to enter the appropriate hi-res or multicolor graphics mode.

3. Execute a statement like OPEN 2,8,2,"*filename*,P,R" to open the disk file for reading (input). Again, the ,P suffix specifies a PRG file, and the ,R suffix opens the file for reading.

4. Execute SYS 881. The ML routine loads the graphics data back into the right memory locations.

5. Execute a statement like CLOSE 2 to close the disk file. Again, you should use the same logical file number (2 in this case) used when opening the file.

6. At this point you can continue with a BASIC program or do whatever else you like.

Graphics Demonstrations

Programs 3 and 4 contain practical demonstrations of how to use these two routines from BASIC. Type in and save both programs, then load and run Program 1 to put the ML save routine in memory. Now load and run Program 3. This program uses the hi-res drawing example from pages 123–126 in the *Commodore 64 Programmer's Reference Guide.* Lines 110–140 define the hi-res screen and color memory to start at locations 8192 and 1024, respectively, then clear the graphics screen. Lines 150–230 draw a simple sine wave pattern. (Be patient; it takes a few minutes to complete the drawing.) Line 270 opens the disk file using 2 as the logical file number and SINEWAVE.HIRES as the filename. After checking the disk error channel, the program calls the ML save routine.

The sine wave disappears as the hi-res memory is moved temporarily to a new location and stored in the disk file. Then the routine moves the picture back to its original location, saves color memory, and returns control to BASIC. After checking the error channel again, the BASIC program restores the normal screen display and ends.

Program 4 shows how to use the ML retrieval routine. Since it looks for a file named SINEWAVE.HIRES on the disk, you can run it only after you've used Program 3 to create the file. Run Program 2 to put the ML retrieval routine in memory, then load and run Program 4. Lines 110–130 define the hi-res screen starting at location 24576, a different area than the one it was saved from. Lines 140–150 fill the screen with a uniform pattern. (Note that this is done only for the purpose of demonstration, to confirm that the retrieval routine puts new information on the screen. It is *not* necessary to clear the graphics screen before using this routine.)

Lines 160–190 open a disk file for reading, using the same name as Program 3 (SINEWAVE.HIRES). After checking the error channel (180–190), the retrieval routine is called with SYS 881. The hi-res screen is restored right before your eyes: First the graphics information appears, then color memory is brought in as well. After a brief pause, the program restores the screen to normal and ends.

Inside The ML Routines

The ML save routine saves the currently defined graphics screen and its associated color memory wherever they are located—even if the hi-res screen is stored in the RAM underneath a ROM area. The ML retrieval routine brings the stored screen back into whatever area you have currently defined as the graphics screen, even if that's a different location from the area from which it was saved. This lets you create and store a complex graphics display using one particular graphics aid (Simons' BASIC, etc.) and retrieve it for use by any other program.

Since sprites are independent of other graphics, these routines can't store or retrieve sprite shapes that appear on the screen.

To make this routine compatible with as many programs as possible, memory usage is restricted to three areas. It uses memory locations 679–738 (normally unused) and 828–1023 (the cassette buffer) to store the routines, and also zero-page locations 2 and 251–254. To save a screen, the ML routine first looks in locations 56576 and 53272 to locate the graphics screen and normal screen memory (which becomes the hi-res color memory). It then swaps the 8K bytes of hi-res RAM memory with the contents of locations 24576–32767 ($6000–$7FFF). This is done by "turning off" the computer's ROM chips temporarily so the swapping routine can see hi-res memory no matter where it's located. Then the routine switches the Kernal ROM back in to write the 8,000 bytes of hi-res information to the disk file, and moves the hi-res screen back to its original location. The 1,000-byte screen memory area is written directly to the disk file.

Note that since no memory swapping is done for color memory, this part of your screen *must* be located in a memory area that's not normally hidden by ROM.

Finally, the normal color memory at 55296-56295 (used in multicolor mode) as well as the screen background byte at 53281 is written to disk. The final disk file is 10,003 bytes (40 blocks) long. Two extra bytes are added at the beginning of the file to make it compatible with version 3.0 of the Okimate Color Print program.

The retrieval routine works in reverse, finding where the graphics screen and color memory are located in the current configuration, then restoring everything to the correct memory locations. Since RAM can be POKEd even if it's under ROM, no memory swapping is required and the contents of the disk file are moved directly into the appropriate memory areas.

For instructions on entering these listings, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

Program 1: Screen Saver 64

```
CS 100 REM PROGRAM 1 SCREEN SA
       VER 64
PE 110 SU=0:FORI=688TO738:READ
       X:SU=SU+X:POKEI,X:NEXT
CK 120 IFSU<>4855THENPRINT"ERR
       OR IN DATA IN LINES 170
       -220":STOP
QX 130 SU=0:FORI=828TO1022:REA
       DX:SU=SU+X:POKEI,X:NEXT
PQ 140 IFSU<>31598THENPRINT"ER
       ROR IN DATA IN LINES 24
       0-470":STOP
SR 150 PRINT"SCREEN SAVE INSTA
       LLED":END
MS 170 DATA173,0,221,41,3,73,3
       ,10,10,10
QP 180 DATA10,10,10,133,254,17
       3,24,208
QJ 190 DATA41,8,10,10,101,254,
       133,2,105
DA 200 DATA31,141,169,2,173,24
       ,208,41,240
DB 210 DATA74,74,234,234,101,2
       54,141,168
XM 220 DATA2,105,3,141,167,2,9
       6
KB 240 DATA160,0,132,251,132,2
       53,165,2,133
PH 250 DATA252,169,96,133,254,
       120,165,1,72
MG 260 DATA41,253,133,1,177,25
       1,170,177
DA 270 DATA253,145,251,138,145
       ,253,200
RK 280 DATA208,243,230,252,230
       ,254,165
HK 290 DATA254,201,128,208,233
       ,104,133,1
JH 300 DATA88,234,234,234,96,1
       62,2,32,201
MD 310 DATA255,169,0,32,210,25
       5,165,2,32
SX 320 DATA210,255,160,0,132,2
       51,169,96
RK 330 DATA133,252,234,234,177
       ,251,32,210
XS 340 DATA255,165,252,201,127
       ,240,7,200
PX 350 DATA208,242,230,252,208
       ,238,200
MX 360 DATA152,201,64,208,232,
       96,160,0
FC 370 DATA132,251,173,168,2,1
       33,252,177
FA 380 DATA251,32,210,255,165,
       252,205,167
SH 390 DATA2,240,7,200,208,241
       ,230,252
MD 400 DATA208,237,200,152,201
       ,232,208
J8 410 DATA231,160,0,132,251,1
       69,216,133
XS 420 DATA252,177,251,32,210,
       255,165,252
JD 430 DATA201,219,240,7,200,2
       08,242,230
XE 440 DATA252,208,238,200,152
       ,201,232
XC 450 DATA208,232,173,33,208,
       32,210,255
PH 460 DATA32,204,255,96,32,17
       6,2,32,60
MX 470 DATA3,32,113,3,32,60,3,
       32,163,3,96
```

Program 2: Screen Retriever 64

```
PR 100 REM PROGRAM 2 SCREEN RE
       TRIEVE
PE 110 SU=0:FORI=688TO738:READ
       X:SU=SU+X:POKEI,X:NEXT
HE 120 IFSU<>4855THENPRINT"ERR
       OR IN DATA IN LINES 160
       -180":STOP
JQ 130 SU=0:FORI=881TO1004:REA
       DX:SU=SU+X:POKEI,X:NEXT
XK 140 IFSU<>21290THENPRINT"ER
       ROR IN DATA IN LINES 19
       0-250":STOP
HM 150 PRINT"SCREEN RETRIEVE I
       NSTALLED":END
KB 160 DATA173,0,221,41,3,73,3
       ,10,10,10,10,10,10,133,
       254,173,24,208
PD 170 DATA41,8,10,10,101,254,
       133,2,105,31,141,169,2,
       173,24,208,41,240
KP 180 DATA74,74,234,234,101,2
       54,141,168,2,105,3,141,
       167,2,96
PC 190 DATA32,176,2,162,2,32,1
       98,255,32,207,255,32,20
       7,255,234,160,0,132,251
AH 200 DATA165,2,133,252,32,20
       7,255,145,251,165,252,2
       05,169,2,240,7,200,208
QK 210 DATA241,230,252,208,237
       ,200,152,201,64,208,231
       ,160,0,132,251,173,168,
       2
FR 220 DATA133,252,32,207,255,
       145,251,165,252,205,167
       ,2,240,7,200,208,241,23
       0
AB 230 DATA252,208,237,200,152
       ,201,232,208,231,160,0,
       132,251,169,216,133,252
       ,32
JE 240 DATA207,255,145,251,165
       ,252,201,219,240,7,200,
       208,242,230,252,208,238
       ,200
SH 250 DATA152,201,232,208,232
       ,32,207,255,141,33,208,
       32,204,255,96,999
```

Program 3: Screen Saver Demo

```
DG 100 REM PROGRAM 3 SCREEN SA
       VE DEMO
HS 110 BASE=2*4096:POKE53272,P
       EEK(53272)OR8
BH 120 POKE53265,PEEK(53265)OR
       32
8K 130 FOR I=8ASE TO BASE+7999
       :POKEI,0:NEXT
KD 140 FOR I=1024TO2023:POKEI,
       3:NEXT
MS 150 FOR X=0 TO 319 STEP.5
RF 160 Y=INT(90+80*SIN(X/10))
KR 170 CH=INT(X/8)
QC 180 RO=INT(Y/8)
RF 190 LN=YAND7
AR 200 BY=BASE+RO*320+8*CH+LN
GX 210 8I=7-(XAND7)
BX 220 POKE8Y,PEEK(BY)OR(2↑BI)
AP 230 NEXT X
EG 240 POKE1024,16
GR 250 FOR I=1{2 SPACES}TO 100
       0:NEXT{2 SPACES}I
BM 260 OPEN15,8,15
JK 270 OPEN2,8,2,"SINEWAVE.HIR
       ES,P,W"
RJ 280 INPUT#15,Z1,Z2$,Z3,Z4
AG 290 IF Z1<>0 THEN CLOSE2:CL
       OSE15:PRINT Z1;Z2$;Z3;Z
       4:GOTO350
MX 300 SYS 1007
AS 310 CLOSE2
AM 320 INPUT#15,Z1,Z2$,Z3,Z4
RS 330 IF Z1<>0 THEN CLOSE15:P
       RINT Z1;Z2$;Z3;Z4
MH 340 FOR I=1 TO 1000:NEXT I
MM 350 POKE53265,PEEK(53265)AN
       D 223
DF 360 POKE53272,PEEK(53272)
       {2 SPACES}AND 247
DR 370 END
```

Program 4: Screen Retriever Demo

```
XF 100 REM PROGRAM 4 SCREEN RE
       TRIEVE DEMO
PH 110 POKE56578,PEEK(56578) O
       R 3:POKE 56576,(PEEK(56
       576)AND252)OR 2
HD 120 BASE=6*4096:POKE53272,P
       EEK(53272)OR8
FG 130 POKE53265,PEEK(53265)OR
       32
FM 140 FOR I=8ASE TO BASE+7999
       :POKE I,66:NEXT
CS 150 FOR I=16384+1024 TO 163
       84+2023:POKE I,77:NEXT
GE 160 OPEN15,8,15
QJ 170 OPEN2,8,2,"SINEWAVE.HIR
       ES,P,R"
CC 180 INPUT#15,Z1,Z2$,Z3,Z4
GG 190 IF Z1<>0 THEN CLOSE2:CL
       OSE15:PRINT Z1;Z2$;Z3;Z
       4:GOTO230
RS 200 SYS 881
PJ 210 CLOSE2
MC 220 FOR I=1 TO 2500:NEXT I
DF 230 POKE56576,(PEEK(56576)A
       ND252) OR 3
FR 240 POKE53272,PEEK(53272)AN
       D247
HD 250 POKE53265,PEEK(53265)AN
       D223
```

©

Atari FontMaker

Charles Brannon, Program Editor

"FontMaker" simplifies the design of character sets for all text modes on Atari 400/800, XL, and XE computers. Although programmers will find FontMaker a valuable addition to their utility library, nonprogrammers can also benefit. Next month, we show how to use FontMaker to customize Atari SpeedScript*'s special character set. Whether you want Old English or a computer age font, FontMaker has the tools you need to express your creativity. It requires at least 16K RAM; disk drive recommended.*

"FontMaker" is a sophisticated character editor written completely in machine language to work with all Atari text modes. Even though a character set (or *font*) editor is a handy utility for programmers, you don't need to know anything about machine language or programming to have fun with it. We include a simple subroutine that lets you load and merge character sets with your own BASIC programs. And next month, we'll show how to use the special ANTIC 3 character sets with the Atari version of our *SpeedScript* word processor (COMPUTE!, March 1986).

Since FontMaker is written in machine language for speed and compactness, you need to type it in with MLX, our machine language editor. See the MLX article elsewhere in this issue for instructions on typing in and using MLX to enter machine language programs.

When you run MLX, answer the first three screen prompts like this:

```
Starting Address: 12288
Ending Address: 14887
Run/Init Address: 12288
```

Next you'll be asked "Tape or Disk?". Although FontMaker can load as a boot tape, it's much easier to use with a disk drive. If you press D for Disk, you'll be asked "Boot Disk or Binary File?". Press F to select binary file. FontMaker will run from a boot disk, but without DOS, there's no way to save or load character sets. So make sure you select F, since MLX can't convert from a boot disk to a binary file.

At the first screen prompt, 12288:, start typing the data from Program 1. See the MLX article for a list of commands that let you type in a program in several sessions. When you've typed the last line, MLX prompts you for a disk filename. This will be the name under which FontMaker is saved to disk. If you're using Atari DOS 2.0S, 2.5, or 3.0, you may use the filename AUTORUN.SYS. This allows FontMaker to automatically load and run when you turn on the computer with the disk in the drive. Be sure this disk also contains the DOS files.

If you want to prevent loading FontMaker automatically (especially if you haven't finished typing it but want to save your preliminary typing), you can rename AUTORUN.SYS to some other name. You can also load FontMaker from DOS 2.0S or 2.5 with menu selection L. If you're using another DOS, such as OS/A+ or DOS XL, you may be able to save FontMaker with a different name, such as FONT.COM and type the command FONT to run FontMaker.

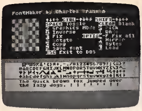
Editing Characters

Assuming you've typed in and saved FontMaker, run it and follow along with this article. You'll see a screen with a colorful 8 × 8 grid, a list of brief instructions, and four rows of characters at the bottom of the screen (see screen photo).

When FontMaker starts, it asks you to pick a character. You can edit one character at a time. When you're asked to select a character, you can use the joystick to move a cursor around in the four rows of characters, then press the fire button to select the character highlighted by the cursor. Or you can simply press the keyboard key corresponding to the character. Action then shifts to the 8 × 8 character editing grid.

Within the grid, you can move the editing cursor (a hollow white box) with either the joystick or the cursor keys. You don't need to use CTRL with the cursor keys to move the cursor—CTRL-cursor up/down/ left/right and SHIFT-cursor up/down/left/right are reserved for other features. When drawing in the character grid, press the fire button or the space bar to reverse (toggle) the dot at the cursor position. Previously set dots are turned off, and blank spaces are turned on. You can hold down the fire button while you move the joystick to draw lines and figures.

As you change the grid, you can see the character in actual size in the character set window. The cursor highlights (reverses) the selected character, but a row of the



"Atari FontMaker" lets you design your own custom character fonts for any Atari text mode.

character you're editing is also displayed. In addition, there's a sample of text ("The quick brown fox jumped over the lazy dogs") so you can judge relative character height and spacing.

To create an entirely new character, you may want to start by pressing SHIFT-CLEAR to erase the existing character pattern. This gives you a clean canvas for your design.

Undo Your Mistakes

If you don't like a change you've made, press U to undo all the changes made since you've selected the character. Press U again to undo the undo, restoring the change you've made. You can press F to fix a character, recopying its image from the standard character set stored in the computer's Read Only Memory (ROM). It's important to distinguish between these options: U reverts to the previous character image, F always gives you the ROM image. If you change the letter A to a spaceship, change B to a rocket, then go back to A and change the spaceship to an alien, U switches between the alien and the spaceship; F gives you the pattern for the letter A. Beware that you can't undo the Fix command.

If you hold down the OPTION button while pressing F, the entire character set is fixed, recopied from ROM. This wipes out any changes you have made to the character set, so be careful.

To select another character to edit, press P and use the joystick or keyboard to pick the new character. In some text modes (see the G command below), the joystick cursor may seem to move strangely. It consistently moves up or down between rows of 32 characters. Since there are only 20 characters per line in modes 1 and 2, these rows wrap around the right margin. You move left and right within a row of characters, and up and down between rows of 32 characters.

FontMaker can display the character set in all Atari text modes. These are GRAPHICS 0 (the default text mode), "GRAPHICS 0½" (technically known as ANTIC 3, a nine-line true-descender mode), multicolor ANTIC mode 4, multicolor ANTIC mode 5, GRAPHICS 1, and GRAPHICS 2. Press G to cycle through these modes.

The *SpeedScript* Character Mode

If you've never heard of the ANTIC text modes, don't fret; they're not normally accessible from BASIC. The Atari *SpeedScript* word processor (and, incidentally, the *PaperClip* word processor from Batteries Included) uses the ANTIC 3 mode for large, readable characters. ANTIC 3, nicknamed GRAPHICS 0½, is a special 40-column mode that lets you define characters within an 8 × 10 character space. Other Atari text modes have only an 8 × 8 character grid. This means that ANTIC 3 characters can have true descenders. (A *descender* is the part of a character that drops below the line of type, such as the tail on a lowercase y or j.)

You still use just eight rows to define a character in ANTIC 3, but the character is positioned within ten screen scan lines. For uppercase characters, the eight rows of the character grid are displayed in rows 1–8 of the character matrix, with two blank lines at the bottom of each character, reserving space for lowercase descenders. For lowercase characters, the first two lines of the character grid are forced blank. The first two rows of the character definition are actually displayed at lines 9 and 10 of the character space, making it easy to reserve space for the descenders. The third through eighth rows of the character are displayed starting at the third line of the character space.

This may sound confusing, but fortunately FontMaker lets you design ANTIC 3 characters without having to mentally translate what you see on the grid to what the character should look like. (However, the Rotate option seems to work strangely in this mode due to the unusual memory configuration used by ANTIC 3 characters.) Just be aware that lowercase characters are actually written two lines lower on the screen than uppercase characters. If you start with the normal character set, you'll need to use the roll and shift options to align the characters. Using the roll or shift commands (see below), roll the lowercase characters down two lines and all the uppercase characters down by one line. While you work, refer to the "quick brown fox" sentence to see that all the characters line up properly.

Since there are forced blank lines for uppercase characters, you can use the lower seven lines to define a large character. Leave the top line blank if you want two-line descenders. The normal Atari character set only uses six columns for a character, giving two pixel spaces between each character on the screen. A larger character can use up to seven columns, leaving the last column blank to keep characters from running into each other. When designing some character sets, though, such as a cursive script, you may want characters to connect together, so you can use the full horizontal space.

You can press the CTRL-cursor keys to *roll* the character within the grid. Pixels that are pushed off the edge of the grid wrap around to the opposite side. This is especially useful for those ANTIC 3 characters. If you press SHIFT with the cursor keys, the pixels that are shifted off the edge of the grid are lost. This can be used to crop a character, or quickly erase a certain column or row. Just roll the character until the column or row you want to erase is at the edge of the grid, then shift the character to push away the pixels.

Other special commands: Press I to invert the character, reversing all the pixels within the grid. R rotates the characters 90 degrees (turns the pattern on its side). Press R twice to turn a character upside-down. M gives you a left-to-right mirror image of the character, as if you picked up the character, flipped it on its back, and put it back down.

Press C to copy the character you're editing to a new position in the character set. The current character replaces the character you select with the joystick or keyboard. Be careful with this, since you can unintentionally erase a cherished character. After the copy, the character you copied to is selected for editing.

Saving And Loading

When you're ready to save your character set, just press S, then type in a legal Atari filename at the prompt:

Save (Device:Filename>

Include the D: for disk or C: for cassette. You can't save to the E: (screen editor) or S: (screen) devices, so if you forget the drive specification and your filename begins with E or S, FontMaker cancels the save. If you selected the save command by mistake and want to cancel it, just press RETURN when asked for the filename.

To load a character set into FontMaker, press L and enter the filename you used to save the character set. This will replace the character set in memory, so be sure to save the one you're working on if you want to keep it. Again, if you select this command by mistake, press RETURN at the prompt to cancel the load.

FontMaker is compatible with character sets created with "SuperFont" (originally published in COMPUTE!, January 1982, and also found in the *First Book of Atari Graphics*). FontMaker sets are also compatible with many other Atari character editors. We found that a FontMaker set can be loaded into the popular *Instedit* editor if you use a filename extension of .SET. Character sets created with the Iridis *Fontedit* also work with FontMaker.

If FontMaker can't save or load a character set due to a bad filename or problem with the disk, it displays the message I/O ERROR: PRESS RETURN. Press RETURN and try the save again, after you've figured out what went wrong.

When you're done editing characters, press the ESC (escape) key to exit FontMaker to DOS. Be sure you've saved the character set you're working on if you want to keep it.

Using Fonts With BASIC

The Bytes option in FontMaker is primarily for programmers who want a list of the eight numbers that define a character, handy for changing just a single character in a program. But if you want to include an entire character set in your program, copying down these numbers can be tedious. Instead, you can use the two BASIC programs below, Programs 1 and 2, to add redefined character sets to BASIC.

Program 2, "Fontloader," is a subroutine you can merge with your BASIC program to load the character set into memory from disk. Just change the filename in line 1010 to the filename of your character set. The program loads the character set at the memory location CHSET, which is defined as eight pages back from the top of memory (location 106). Change this if you want to put your character set somewhere else. If a GRAPHICS command resets the character set, you can use POKE 756,CHSET/256 to re-point the character set vector to your RAM character set. Use POKE 756,224 to display the ROM character set, located at memory location 57344 ($E000).

If you're using a *Translator* type of program on XL and XE computers, you can change CHSET (line 1030) to 57344 to load the character set directly into the RAM space corresponding to the position of the ROM character set. This keeps the character set in memory until you turn off the machine.

Use Program 3, "Character Set Datamaker," if you'd rather store your character set as part of your program in DATA statements. The Datamaker actually creates another program that you can merge with your own program. It creates DATA statements for only those characters that have been changed from the ROM image. Datamaker asks for the filename of your character set and a filename you'd like to use for the program it creates.

After using Datamaker, type NEW and use ENTER to load the program created by Datamaker. As with Program 2, you can change CHSET in the program created by Datamaker if you want to determine yourself where the character set should go in memory. The program created by Datamaker copies the ROM set down to RAM (at CHSET), POKEs in the new characters, and switches the character pointer at 756 to the new set. Again, you can use POKE 756, CHSET/256 to reenable the set if your program somehow resets this pointer.

Next month we'll provide a program that lets you install an ANTIC 3 character set into your copy of *SpeedScript*, along with tips for creating readable characters and for using special characters for printer effects.

Program 1: FontMaker

Please refer to the "MLX" article in this issue before entering the following listing

```
12288:169,125,032,096,054,032,252
12294:187,053,032,009,054,032,117
12300:062,050,032,108,050,032,090
12306:045,048,169,001,141,194,104
12312:048,032,242,048,032,113,027
12318:048,032,017,049,076,083,079
12324:051,169,064,160,048,032,048
12330:130,054,096,032,037,048,183
12336:169,012,133,082,169,137,238
12342:160,052,032,145,054,169,154
12348:002,133,082,096,070,111,042
12354:110,116,077,097,107,101,162
12360:114,032,098,121,032,067,024
12366:104,097,114,108,101,115,205
12372:032,066,114,097,110,110,101
12378:111,110,000,173,202,048,222
12384:201,003,208,010,173,194,117
12390:048,201,096,144,003,169,251
12396:000,096,169,001,096,032,246
12402:093,048,208,006,032,202,191
12408:054,032,202,054,169,128,247
12414:009,024,133,203,173,198,098
12420:048,009,002,133,204,162,178
12426:000,160,000,189,169,048,192
12432:145,203,200,152,041,003,120
12438:208,245,232,224,008,208,251
12444:240,032,093,048,208,006,015
12450:032,252,054,032,252,054,070
12456:096,000,000,000,000,000,008
12462:000,000,000,000,000,000,174
12468:000,000,000,000,000,000,180
12474:000,000,000,000,000,000,186
12480:000,000,000,000,000,000,192
12486:000,000,000,000,002,000,200
12492:000,000,000,000,169,000,117
12498:133,204,173,194,048,010,204
12504:038,204,010,038,204,010,208
12510:038,204,024,133,207,109,169
12516:195,048,133,203,165,204,152
12522:133,208,109,196,048,133,037
12528:204,096,032,208,048,160,220
12534:007,177,203,153,169,048,235
12540:153,177,048,136,016,245,003
12546:096,032,208,048,160,007,041
12552:185,169,048,145,203,136,126
12558:016,248,096,165,088,024,139
12564:105,228,133,203,165,089,175
12570:105,001,133,204,169,000,126
12576:141,193,048,162,004,173,241
12582:202,048,201,006,144,002,129
12588:162,002,160,000,173,193,222
12594:048,205,194,048,208,002,243
12600:009,128,145,203,238.193,204
12606:048,200,192,032,208,236,210
12612:024,165,203,105,040,133,226
12618:203,165,204,105,000,133,116
12624:204,202,208,218,173,202,007
12630:048,201,005,176,014,160,178
12636:000,185,119,049,201,010,144
12642:240,007,145,203,200,208,077
12648:244,160,000,200,173,194,051
```

```
12654:048,145,203,200,192,070,200
12660:144,245,096,052,104,101,090
12666:000,113,117,105,099,107,151
12672:000,098,114,111,119,110,168
12678:000,102,111,120,000,106,061
12684:117,109,112,101,100,000,167
12690:111,118,101,114,000,000,078
12696:000,000,000,000,000,000,152
12702:000,116,104,101,000,108,075
12708:097,122,121,000,100,111,203
12714:103,115,014,010,000,000,156
12720:000,000,000,033,225,001,179
12726:000,031,223,255,000,032,211
12732:224,000,080,105,099,107,035
12738:032,097,032,099,104,097,143
12744:114,097,099,116,101,114,073
12750:046,046,046,000,169,190,191
12756:160,049,032,130,054,032,157
12762:017,049,032,242,048,032(126
12768:113,048,173,132,002,240,164
12774:042,173,252,002,201,255,131
12780:208,043,174,120,002,189,204
12786:174,049,240,236,032,112,061
12792:053,142,031,208,024,109,047
12798:194,048,041,127,174,202,016
12804:048,224,006,144,002,041,213
12810:063,141,194,048,076,217,237
12816:049,173,132,002,240,251,095
12822:076,059,050,032,029,058,070
12828:041,127,201,096,176,013,170
12834:201,032,176,006,024,105,066
12840:064,076,047,050,056,233,054
12846:032,141,194,048,032,017,254
12852:049,032,242,048,032,113,056
12858:048,076,037,048,165,106,026
12864:056,233,016,133,204,141,079
12870:196,048,169,000,133,203,051
12876:141,195,048,141,194,048,075
12882:133,205,169,224,133,206,128
12888:160,000,177,205,145,203,210
12894:200,208,249,230,204,230,135
12900:206,165,206,201,229,208,035
12906:239,096,169,046,141,047,076
12912:002,169,003,141,029,208,152
12918:169,001,141,111,002,173,203
12924:196,048,056,233,004,141,034
12930:198,048,141,007,212,169,137
12936:000,141,197,048,162,011,183
12942:157,255,207,202,208,250,141
12948:173,198,048,133,204,173,053
12954:197,048,133,203,162,004,133
12960:160,000,152,145,203,200,252
12966:208,251,230,204,202,208,189
12972:246,024,169,128,133,209,057
12978:169,000,133,203,173,198,030
12984:048,009,003,133,204,133,202
12990:210,169,170,133,207,160,215
12996:024,165,207,145,203,073,245
13002:255,145,209,200,152,041,180
13008:003,208,006,165,207,073,102
13014:255,133,207,192,056,208,241
13020:232,169,056,141,001,208,003
13026:141,002,208,141,003,208,161
13032:169,003,141,009,208,141,135
13038:010,208,141,011,208,169,217
13044:014,141,192,002,169,090,084
13050:141,193,002,169,200,141,072
13056:194,002,169,038,141,195,227
13062:002,096,173,200,048,141,154
13068:000,208,173,198,048,009,136
13074:002,133,204,160,000,152,157
13080:170,133,203,145,203,200,054
13086:016,251,172,199,048,189,137
13092:047,051,145,203,200,232,146
13098:224,004,208,245,096,240,035
13104:144,144,240,000,000,000,064
13110:000,000,001,001,001,000,057
13116:255,255,255,000,000,000,057
13122:000,000,000,000,000,000,066
13128:001,255,000,000,001,255,072
13134:000,000,001,255,000,169,247
13140:056,141,200,048,169,024,210
13146:141,199,048,032,008,051,057
13152:174,120,002,142,201,048,015
13158:224,015,240,064,169,004,050
13164:133,207,141,031,208,165,225
13170:020,197,020,240,252,174,249
13176:201,048,189,051,051,024,172
13182:109,200,048,201,056,144,116
13188:007,201,085,176,003,141,233
13194:200,048,189,067,051,024,205
13200:109,199,048,201,024,144,101
13206:007,201,053,176,003,141,219
13212:199,048,032,008,051,198,180
13218:207,165,207,208,202,032,159
13224:112,053,173,252,002,201,193
13230:255,208,085,173,132,002,005
13236:200,170,056,173,199,048,010
13242:233,024,074,074,168,032,023
13248:093,048,208,014,192,006,241
13254:176,005,200,200,076,210,041
```

```
13260:051,152,056,233,006,168,102
13266:173,200,048,056,233,056,208
13272:074,074,170,232,056,169,223
13278:000,106,202,208,252,089,055
13284:169,048,153,169,048,032,079
13290:167,054,032,016,055,173,219
13296:201,048,201,015,208,005,150
13302:162,200,032,114,053,076,115
13308:093,051,169,000,141,000,194
13314:208,076,037,048,032,029,176
13320:058,201,097,144,007,201,204
13326:123,176,003,056,233,032,125
13332:174,058,052,221,058,052,123
13338:240,006,202,208,248,076,238
13344:096,051,202,138,010,170,187
13350:169,051,072,169,092,072,151
13356:189,086,052,072,189,085,205
13362:052,072,169,000,141,000,228
13368:208,096,026,045,061,043,023
13374:042,028,029,030,031,032,254
13380:027,080,125,095,124,092,099
13386:094,071,073,085,070,077,032
13392:082,083,076,067,066,092,034
13398:055,099,055,106,055,113,057
13404:055,194,054,244,054,060,241
13410:055,076,055,181,051,121,125
13416:055,209,049,142,055,173,019
13422:054,223,054,036,055,048,068
13428:055,155,053,156,055,173,251
13434:055,196,055,005,056,028,005
13440:056,183,056,246,056,185,142
13446:057,218,057,155,155,027,035
13452:028,027,029,027,030,027,052
13458:031,032,195,212,210,204,006
13464:045,027,028,027,029,027,079
13470:030,027,031,032,211,200,177
13476:201,198,212,045,027,028,107
13482:027,029,027,030,027,031,085
13488:155,211,208,193,195,197,055
13494:032,084,111,103,103,108,211
13500:101,032,032,032,032,195,100
13506:204,197,193,210,032,066,072
13512:108,097,110,107,155,199,208
13518:032,071,114,097,112,104,224
13524:105,099,115,032,077,111,239
13530:100,101,032,208,032,080,003
13536:105,099,107,155,201,032,155
13542:073,110,118,101,114,115,093
13548:101,032,032,032,032,032,241
13554:032,032,213,032,085,110,234
13560:100,111,155,198,032,070,146
13566:105,120,032,032,032,032,095
13572:032,032,207,208,212,201,128
13578:207,206,045,070,032,070,128
13584:105,120,032,097,108,108,074
13590:155,210,032,082,111,116,216
13596:097,116,101,032,032,032,182
13602:032,032,032,032,032,205,143
13608:032,077,105,114,114,111,081
13614:114,155,195,032,067,111,208
13620:112,121,032,032,032,032,157
13626:032,032,032,032,032,032,250
13632:194,032,066,121,116,101,182
13638:115,155,211,032,083,097,251
13644:118,101,032,102,111,110,138
13650:116,032,032,032,032,032,102
13656:204,032,076,111,097,100,196
13662:155,197,211,195,032,069,185
13668:120,105,116,032,116,111,188
13674:032,068,079,083,000,000,112
13680:162,090,160,000,136,208,100
13686:253,202,208,250,096,024,127
13692:165,088,105,228,133,203,022
13698:133,209,165,089,105,001,064
13704:133,204,133,210,162,002,212
13710:169,000,160,145,203,200,003
13716:208,251,230,204,202,208,171
13722:246,096,032,123,053,238,174
13728:202,048,173,202,048,201,010
13734:008,144,005,169,002,141,123
13740:202,048,032,187,053,032,214
13746:242,048,032,113,048,032,181
13752:017,049,096,173,048,002,057
13758:133,203,173,049,002,133,115
13764:204,160,003,169,194,145,047
13770:203,160,016,169,134,145,005
13776:203,200,169,006,145,203,110
13782:200,173,202,048,145,203,161
13788:200,192,022,208,249,201,012
13794:006,240,004,201,005,176,090
13800:017,169,016,145,203,173,187
13806:202,048,200,145,203,200,212
13812:145,203,200,145,203,200,060
13818:169,065,145,203,165,203,176
13824:200,145,203,165,204,200,093
13830:145,203,096,169,064,141,056
13836:014,212,169,030,141,000,066
13842:002,169,054,141,001,002,131
13848:169,192,141,014,212,096,080
13854:072,138,072,173,011,212,196
13860:201,032,176,011,169,000,113
13866:141,010,212,141,024,208,010
```

```
13072:076.087.054.141,010,212,116
13878:173,202,048,201,004,144,058
13884:009,173,198,002,141,024,095
13890:208,076,081,054,162,005,140
13896:189,090,054,157,021,208,023
13902:202,208,247,173,196,048,128
13908:141,009,212,104,170,104,056
13914:064,076,002,010,254,008,248
13920:140,203,048,162,000,142,023
13926:072,003,142,073,003,160,043
13932:011,140,066,003,032,086,190
13938:228,172,203,048,096,160,253
13944:000,152,145,088,200,192,129
13950:040,208,249,096,072,152,175
13956:072,169,002,133,085,032,113
13962:119,054,133,084,104,168,032
13968:104,133,203,132,204,169,065
13974:001,141,240,002,160,000,182
13980:177,203,240,006,032,096,142
13986:054,200,208,246,096,032,230
13992:003,049,032,113,048,096,253
13998:032,202,054,162,007,032,151
14004:093,048,208,002,162,001,182
14010:169,000,157,169,048,032,249
14016:167,054,096,032,202,054,029
14022:032,167,054,096,173,169,121
14028:048,072,162,000,189,170,077
14034:048,157,169,048,232,224,064
14040:007,208,245,104,141,176,073
14046:048,096,032,252,054,162,098
14052:000,032,093,048,208,002,099
14058:162,002,169,000,157,169,125
14064:048,032,167,054,096,032,1[illegible]7
14070:252,054,032,167,054,096,135
14076:173,176,048,072,162,007,122
14082:189,168,048,157,169,048,013
14088:202,208,247,104,141,169,055
14094:048,096,169,020,141,000,232
14100:210,162,175,142,001,210,152
14106:160,128,136,208,253,202,089
14112:224,159,208,243,096,162,100
14118:007,030,169,048,202,208,190
14124:250,032,167,054,096,162,037
14130:007,094,169,048,202,208,010
14136:250,032,167,054,096,162,049
14142:007,189,169,048,010,062,035
14148:169,048,202,016,246,032,013
14154:167,054,096,162,007,189,237
14160:169,048,074,126,169,048,202
14166:202,016,246,032,167,054,035
14172:096,104,104,162,014,076,136
14178:099,051,104,104,162,013,119
14184:076,099,051,104,104,162,188
14190:011,076,099,051,104,104,043
14196:162,007,076,099,051,096,095
14202:169,064,141,014,212,169,123
14208:000,141,029,208,162,003,159
14214:157,000,208,202,016,250,199
14220:108,010,000,169,000,162,077
14226:007,157,169,048,202,016,233
14232:250,032,167,054,096,162,145
14238:007,189,169,048,073,255,131
14244:157,169,048,202,016,245,233
14250:032,167,054,096,162,007,176
14256:189,169,048,072,189,177,252
14262:048,157,169,048,104,157,097
14268:177,048,202,016,239,032,134
14274:167,054,096,169,008,141,061
14280:031,208,173,031,208,041,124
14286:007,201,003,208,018,032,163
14292:062,050,169,001,141,194,061
14298:048,032,242,048,032,017,125
14304:049,032,113,048,096,032,082
14310:208,048,165,207,133,203,170
14316:165,208,024,105,224,133,071
14322:204,162,000,160,000,177,177
14328:203,157,169,048,200,232,233
14334:224,008,208,245,032,167,114
14340:054,096,162,007,160,008,235
14346:169,000,094,169,048,042,020
14352:136,208,249,157,169,048,215
14358:202,016,239,032,167,054,220
14364:096,160,000,162,007,169,110
14370:000,030,169,048,042,202,013
14376:016,249,153,185,048,200,123
14382:192,008,208,237,162,007,092
14388:189,185,048,157,169,048,080
14394:202,016,247,032,167,054,008
14400:096,160,000,140,240,002,190
14406:169,032,032,096,054,169,110
14412:126,032,096,054,140,204,216
14418:048,032,029,058,172,204,113
14424:048,201,155,240,047,201,212
14430:126,240,032,041,127,201,093
14436:032,144,233,201,097,144,183
14442:007,201,123,176,003,056,160
14448:233,032,166,085,224,039,123
14454:240,216,153,128,005,032,124
14460:096,054,200,208,207,192,057
14466:000,240,203,032,096,054,243
14472:136,076,080,056,162,001,135
```

```
14478:142,24Ø,ØØ2,Ø32,Ø96,Ø54,196
14484:169,ØØØ,153,128,ØØ5,14Ø,231
1449Ø:2Ø4,Ø48,173,128,ØØ5,Ø96,Ø4Ø
14496:Ø83,Ø97,118,1Ø1,Ø32,Ø4Ø,119
145Ø2:Ø68,1Ø1,118,1Ø5,Ø99,1Ø1,246
145Ø8:Ø58,Ø7Ø,1Ø5,1Ø8,1Ø1,11Ø,212
14514:Ø97,1Ø9,1Ø1,Ø41,Ø62,ØØØ,Ø76
1452Ø:169,16Ø,16Ø,Ø56,Ø32,13Ø,123
14526:Ø54,Ø32,Ø65,Ø56,24Ø,Ø24,149
14532:2Ø1,Ø69,24Ø,Ø2Ø,2Ø1,Ø83,242
14538:24Ø,Ø16,Ø32,111,Ø57,169,Ø59
14544:ØØ8,141,2Ø5,Ø48,169,Ø11,Ø22
1455Ø:141,2Ø6,Ø48,Ø32,Ø3Ø,Ø57,216
14556:Ø76,Ø37,Ø48,Ø76,111,Ø97,153
14562:1ØØ,Ø32,Ø4Ø,Ø68,1Ø1,118,173
14568:1Ø5,Ø99,1Ø1,Ø58,Ø7Ø,1Ø5,ØØ2
14574:1Ø8,1Ø1,11Ø,Ø97,1Ø9,1Ø1,Ø96
1458Ø:Ø41,Ø62,ØØØ,169,223,16Ø,131
14586:Ø56,Ø32,13Ø,Ø54,Ø32,Ø65,1Ø7
14592:Ø56,24Ø,Ø24,2Ø1,Ø69,24Ø,Ø62
14598:Ø2Ø,2Ø1,Ø83,24Ø,Ø16,Ø32,Ø86
146Ø4:111,Ø57,169,ØØ4,141,2Ø5,187
1461Ø:Ø48,169,ØØ7,141,2Ø6,Ø48,125
14616:Ø32,Ø3Ø,Ø57,Ø76,Ø37,Ø48,Ø48
14622:169,128,157,Ø68,ØØ3,169,212
14628:ØØ5,157,Ø69,ØØ3,173,2Ø4,135
14634:Ø48,157,Ø72,ØØ3,169,ØØØ,235
1464Ø:157,Ø73,ØØ3,169,ØØ3,157,Ø98
14646:Ø66,ØØ3,173,2Ø5,Ø48,157,194
14652:Ø74,ØØ3,169,ØØØ,157,Ø75,Ø26
14658:ØØ3,Ø32,Ø86,228,Ø48,Ø5Ø,ØØ1
14664:173,195,Ø48,157,Ø68,ØØ3,2Ø4
1467Ø:173,196,Ø48,157,Ø69,ØØ3,212
14676:169,ØØØ,157,Ø72,ØØ3,169,142
14682:ØØ4,157,Ø73,ØØ3,173,2Ø6,194
14688:Ø48,157,Ø66,ØØ3,Ø32,Ø86,232
14694:228,Ø48,Ø17,Ø32,111,Ø57,Ø83
147ØØ:Ø48,Ø12,Ø96,162,Ø16,169,Ø99
147Ø6:Ø12,157,Ø66,ØØ3,Ø32,Ø86,214
14712:228,Ø96,Ø32,111,Ø57,169,Ø45
14718:136,16Ø,Ø57,Ø32,13Ø,Ø54,183
14724:Ø32,Ø29,Ø58,Ø96,Ø73,Ø47,211
1473Ø:Ø79,Ø32,Ø69,Ø82,Ø82,Ø79,Ø49
14736:Ø82,Ø58,253,Ø32,Ø8Ø,Ø82,219
14742:Ø69,Ø83,Ø83,Ø32,21Ø,197,Ø56
14748:212,213,21Ø,2Ø6,ØØØ,Ø67,Ø4Ø
14754:111,112,121,Ø32,116,111,253
1476Ø:Ø32,119,1Ø4,1Ø5,Ø99,1Ø4,219
14766:Ø32,Ø99,1Ø4,Ø97,114,Ø97,2Ø5
14772:Ø99,116,1Ø1,114,Ø63,ØØØ,161
14778:162,ØØ7,189,169,Ø48,157,15Ø
14784:185,Ø48,2Ø2,Ø16,247,169,Ø35
1479Ø:161,16Ø,Ø57,Ø32,214,Ø49,1Ø3
14796:162,ØØ7,189,185,Ø48,157,184
148Ø2:169,Ø48,2Ø2,Ø16,247,Ø32,156
148Ø8:167,Ø54,Ø96,Ø32,119,Ø54,226
14814:162,ØØØ,134,Ø85,142,2Ø7,184
1482Ø:Ø48,189,169,Ø48,Ø32,25Ø,196
14826:Ø57,174,2Ø7,Ø48,232,224,152
14832:ØØ8,2Ø8,239,Ø32,Ø29,Ø58,Ø46
14838:Ø32,Ø37,Ø48,Ø96,133,212,Ø36
14844:169,ØØØ,133,213,Ø32,17Ø,2Ø1
1485Ø:217,Ø32,23Ø,216,16Ø,ØØØ,Ø89
14856:177,243,Ø48,ØØ6,Ø32,Ø96,Ø98
14862:Ø54,2ØØ,2Ø8,246,Ø41,127,122
14868:Ø32,Ø96,Ø54,169,Ø32,Ø32,179
14874:Ø96,Ø54,Ø96,173,Ø37,228,198
1488Ø:Ø72,173,Ø36,228,Ø72,162,ØØ7
14886:ØØØ,Ø96,224,ØØ2,225,ØØ2,Ø75
```

Program 2: Fontloader

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
JD 1ØØØ REM FDNTLDADER
FH 1Ø1Ø OPEN #1,4,Ø,"D:SERIF
        .SET":REM YDUR FILEN
        AME HERE
HD 1Ø2Ø X=16:REM FILENUM*16
MD 1Ø25 DIM CID$(7):CIO$="hh
        h":CID$(4)=CHR$(17Ø)
        :CID$(5)="LV":CID$(7
        )=CHR$(228)
OD 1Ø3Ø CHSET=(PEEK(1Ø6)-8)*
        256:PDKE 756,CHSET/2
        56:REM ADDRESS DF CH
        ARACTER SET.  TRY 57
        344 DN XL'S WITH TRA
        NSLATDR
IJ 1Ø4Ø ICCDM=834:IC8ADR=836
        :IC8LEN=84Ø
NF 1Ø5Ø POKE ICBADR+X+1,CHSE
        T/256:POKE IC8ADR+X,
        Ø
NH 1Ø6Ø POKE ICBLEN+X+1,4:PD
        KE ICBLEN+X,Ø
PG 1Ø7Ø PDKE ICCDM+X,7:A=USR
        (ADR(CID$),X)
JD 1Ø8Ø CLOSE #1
MB 1Ø9Ø RETURN :REM REMDVE T
        HIS LINE TD USE THIS
         AS A STAND-ALDNE PR
        DGRAM
```

Program 3: Character Set Datamaker

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!.

```
PE 1ØØ OPEN #1,12,Ø,"E:"
KH 1Ø2 GRAPHICS 1+16
HI 1Ø5 DIM F$(14),DF$(14),T$
       (12),A(7)
OA 11Ø PDSITIDN 3,Ø:? #6;"ch
       aracter set"
MI 12Ø PDSITIDN 5,2:? #6;"DA
       TAMAKER":? #6
FJ 13Ø ? #6;"THIS UTILITY CR
       EATES";
HB 14Ø ? #6;"A SET DF DATA S
       TATE-";
BN 15Ø ? #6;"MENTS FRDM A SA
       VED"
PC 16Ø ? #6;"CHARACTER SET.
        IT"
FA 17Ø ? #6;"DPTIMIZES 8Y ON
       LY"
KK 18Ø ? #6;"LISTING CHARACT
       ERS"
DL 19Ø ? #6;"NDT PRESENT IN
       THE"
IH 2ØØ ? #6;"STANDARD CHARAC
       TER"
ME 21Ø ? #6;"SET."
KD 22Ø ? #6:? #6;"PRESS OPTI
       ON"
BF 23Ø IF PEEK(53279)<>3 THE
       N 23Ø
KK 24Ø GRAPHICS 1+16
LB 25Ø ? #6;"THE DATA STATEM
       ENTS"
DO 26Ø ? #6;"WILL 8E WRITTEN
        TD"
IM 27Ø ? #6;"DISK AS A I1st
       FILE"
MP 28Ø ? #6;"USE enter TD ME
       RGE"
DI 29Ø ? #6;"THE DATA WITH Y
       DUR"
JB 3ØØ ? #6;"PRDGRAM.":? #6:
       ? #6;"ENTER FILENAME"
       :? #6;"OF CHARACTER S
       ET"
MB 3Ø5 PDKE 82,Ø:PDKE 87,Ø
GG 31Ø ? CHR$(28);CHR$(156);
       "[illegible]";:INPUT #1;T$:IF
       T$="" THEN 31Ø
NP 315 F$="D:":F$(3)=T$
PP 32Ø ? CHR$(125);"ENTER OU
       TPUT{8 SPACES}FILENAM
       E":? :?
GK 33Ø ? CHR$(28);CHR$(156);
       "[illegible]";:INPUT #1;T$:IF
       T$="" THEN 33Ø
HM 332 DF$="D:":DF$(3)=T$
AH 335 ? CHR$(125);"ENTER LI
       NE # FOR{5 SPACES}DAT
       A STATEMENTS":? :?
KC 34Ø INPUT SLINE
BG 345 CLDSE #1
AH 35Ø GRAPHICS 2+16:PDSITID
       N 5,6:? #6;"working
       {3 N}":SETCDLDR 4,3,4
HC 37Ø DPEN #1,4,Ø,F$
DD 38Ø TRAP 6ØØ:DPEN #2,8,Ø,
       DF$:TRAP 4ØØØØ
BG 381 ? #2;SLINE;"CHSET=(PE
       EK(1Ø6)-8)*256:FDR I=
       Ø TD 1Ø23:POKE CHSET+
       I,PEEK(57344+I):NEXT
       I"
PM 382 ? #2;SLINE+1;"RESTDRE
        ";SLINE+5
NF 383 ? #2;SLINE+2;"READ A:
       IF A=-1 THEN RETURN"
CK 384 ? #2;SLINE+3;"FDR J=Ø
        TD 7:READ 8:PDKE CHS
       ET+A*8+J,8:NEXT J"
BE 385 ? #2;SLINE+4;"GDTO ";
       SLINE+2
DB 387 LINE=SLINE+4
BD 39Ø FDR I=Ø TD 127:F=Ø
AM 4ØØ FDR J=Ø TO 7
MK 41Ø GET #1,A:A(J)=A
IL 42Ø IF A<>PEEK(57344+I*8+
       J) THEN F=1
CA 43Ø NEXT J
CH 44Ø IF  NDT F THEN 46Ø
IG 445 LINE=LINE+1
PP 45Ø ? #2;LINE;" DATA ";:?
        #2;I;:FDR J=Ø TD 7:?
        #2;",";A(J);:NEXT J:
       ? #2
KB 46Ø NEXT I:? #2;LINE+1;"D
       ATA -1"
CN 47Ø PDKE 82,2:GRAPHICS Ø:
       ? "AI1 finished'  Use
        ENTER ";OF$
CB 48Ø ? "to merge the file.
       "
HE 49Ø END
FL 6ØØ PDKE 82,2:GRAPHICS Ø:
       ? :? "ERRDR TRYING TD
        DPEN ";DF$;"."
GD 61Ø END
```

©

COMPUTE! Subscriber Services

Pleose help us serve you better. If you need to contoct us for ony of the reasons listed below, write to us ot:

COMPUTE! Magazine
P.O. Box 10954
Des Moines, IA 50340

or coll the Toll Free number listed below.

Change Of Address. Pleose ollow us 6–B weeks to effect the chonge; send your current moiling lobel olong with your new oddress.

Renewal. Should you wish to renew your **COMPUTE!** subscription before we remind you to, send your current moiling lobel with poyment or chorge number or coll the Toll Free number listed below.

New Subscription. A one year (12 month) US subscription to **COMPUTE!** is $24.00 (2 yeors, $45.00; 3 yeors, $65.00. For subscription rotes outside the US, see staff page). Send us your nome ond oddress or coll the Toll Free number listed below.

Delivery Problems. If you receive duplicate issues of **COMPUTE!**, if you experience lote delivery or if you have problems with your subscription, please coll the Toll Free number listed below

COMPUTE!
1-800-247-5470
In IA 1-800-532-1272

Hi-Res
Graphics Aid Routines

Jon Hylands

This handy utility makes it easy to perform sophisticated operations on Apple high-resolution graphics screens: inverting screens, copying screens, superimposing one screen on another, and more. It works on any Apple II-series computer with DOS 3.3 or ProDOS.

Like most personal computers, Apple II-series machines can display high-resolution color graphics. There are many commercial programs that let you draw, save, and reload hi-res screens. But few of them let you easily perform complex operations such as inverting an entire hi-res screen or superimposing one screen on another. "Hi-Res Graphics Aid" fills that gap. Though the program uses machine language for speed, you don't need to know ML to use it.

Type in and save the program below, then run it. The screen prompts are self-explanatory. Keep in mind that this is not a general-purpose drawing or design program; it performs large-scale tasks on existing graphics screens. Since the Apple can store two hi-res screens in memory at a time, most operations let you act on either screen 1 or screen 2.

When you run Graphics Aid, it displays a main menu of six selections. From this menu you can display a screen, edit a screen, load a screen, save a screen, display a disk catalog, or quit. The current selection is highlighted in inverse video. To choose a different selection, press the up-arrow or down-arrow keys (CTRL-K or CTRL-J on the Apple II+) and then press RETURN. Here's a brief description of the options:

Display screen. Enter 1 to display screen 1; 2 for screen 2.

Edit screen. This option displays a second menu with the following options:

- *Display screen.* Enter 1 or 2.
- *Invert screen.* Enter 1 or 2.
- *Copy screen.* Enter 1 to copy screen 1 to screen 2, or vice versa.
- *Superimpose screen.* Enter 1 to superimpose screen 1 on screen 2, or vice versa. Then choose the mode by pressing a number key from 1–3. Mode 1 is ORA mode; every pixel that's turned on in either screen remains on. Mode 2 is AND mode; only pixels that are on in both screens remain on. In Mode 3 (XOR), every pixel that's turned on in both screens will be turned off, and vice versa.
- *Color screen.* Choose screen 1 or 2, then enter a color number from 0–7.
- *Flip high bits.* Choose screen 1 or 2, then choose the mode by pressing a number key from 1–3. Mode 1 sets the high bits, mode 2 clears them, and mode 3 inverts them (on bits are turned off, and vice versa).
- *Swap screens.* Swap the contents of screen 1 and screen 2.
- *Return to command menu.*

Load to screen. Choose screen 1 or 2, then select drive 1 or 2 and enter the filename of the graphics file you wish to load.

Save screen. Choose screen 1 or 2, then select drive 1 or 2 and enter the filename you wish to use when saving the graphics screen to disk.

Catalog. Displays a disk catalog.

Quit. Exit to BASIC.

Hi-Res Graphics Aid

For instructions on entering this listing, please refer ta "COMPUTEI's Guide to Typing In Programs" in this issue af COMPUTEI.

```
B2 10 BA = 32768: FDR I = BA TD
      BA + 212: READ A:CK = CK +
       A: PDKE I,A: NEXT : REM L
      DAD HR.CDDE
CA 20 IF CK < > 31397 THEN PRINT
       "ERRDR IN DATA STATEMENTS
      .": STDP
F4 30 DATA 76,18,128,76,33,128,7
      6,55
A0 40 DATA 128,76,80,128,76,115,
      128,76
E1 50 DATA 151,128,166,255,173,8
      0,192,173
E7 60 DATA 82,192,173,87,192,189
      ,83,192
7D 70 DATA 96,166,255,189,195,12
      8,133,251
35 80 DATA 32,186,128,177,250,73
      ,255,145
5F 90 DATA 250,32,177,128,208,24
      5,96,166
68 100 DATA 255,189,195,128,133,
       251,189,198
F9 110 DATA 128,133,253,32,186,1
       28,177,250
BE 120 DATA 145,252,32,177,128,2
       08,247,96
FD 130 DATA 166,255,189,195,128,
       133,251,189
07 140 DATA 198,128,133,253,166,
       254,189,201
05 150 DATA 128,141,105,128,32,1
       86,128,177
```

```
45 160 OATA 25Ø,17,252,145,252,3
       2,177,128
21 17Ø OATA 2ØB,245,96,166,255,1
       B9,195,12B
87 1BØ DATA 133,251,166,254,1B9,
       2Ø5,12B,141
ED 19Ø OATA 141,12B,1B9,2Ø9,12B,
       141,142,12B
CA 2ØØ DATA 32,1B6,12B,177,25Ø,9
       ,12B,145
79 21Ø DATA 25Ø,32,177,12B,2ØB,2
       45,96,169
82 22Ø OATA 32,133,251,1Ø,133,25
       3,32,1B6
92 23Ø DATA 12B,177,25Ø,72,177,2
       52,145,25Ø
88 24Ø DATA 1Ø4,145,252,32,177,1
       2B,2ØB,241
DE 25Ø OATA 96,2ØØ,2ØB,5,23Ø,251
       ,23Ø,253
73 26Ø OATA 2Ø2,96,16Ø,Ø,132,25Ø
       ,132,252
EA 27Ø OATA 162,32,96,Ø,32,64,Ø,
       64
23 2BØ OATA 32,Ø,17,49,B1,Ø,9,41
38 29Ø DATA 73,Ø,12B,127,12B
16 3ØØ TEXT : HOME : PRINT : PRI
       NT CHR$ (4);"PR#Ø": PRINT
        : REM INITIALIZATION
2A 31Ø D$ = CHR$ (4):BE$ = CHR$
       (7):E$ = CHR$ (27):L$ = "
       ----------":O = 1
F4 32Ø REAO L: OIM X(L),TI$(L,13
       )
52 33Ø FDR J = 1 TO L: READ X(J)
       : FDR I = 1 TD X(J): READ
        TI$(J,I): NEXT : NEXT
57 34Ø DATA 2,6,DISPLAY SCREEN,S
       CREEN EDITDR,LDAO SCREEN,
       SAVE SCREEN,CATALOG,OUIT
E4 35Ø OATA B,OISPLAY SCREEN,INV
       ERT SCREEN,CDPY SCREEN,SU
       PERIMPOSE SCREEN,CDLOR SC
       REEN,FLIP HI BITS,SWAP SC
       REENS,COMMAND MENU
18 36Ø READ X: DIM ER$(X): FOR I
        = 1 TD X: REAO ER$(I): N
       EXT
CC 37Ø DATA 13,,,,WRITE PRDTECTE
       D,,FILE NOT FOUND,VOLUME
       MISMATCH,I/D ERRDR,DISK F
       ULL,FILE LOCKEO,SYNTAX ER
       RDR,,FILE TYPE MISMATCH
Ø9 38Ø REM  CDMMAND MENU
81 39Ø HDME : TEXT :J = 1:M$ = "
       HI.RES CDMMAND MENU": GDS
       UB 64Ø
51 4ØØ IF I = X(J) THEN VTAB 1Ø
       + X(J): END
5Ø 41Ø ON I GOSUB B9Ø,B4Ø,129Ø,1
       36Ø,143Ø
9B 42Ø GOTO 39Ø
4B 43Ø REM  GET A KEYSTROKE
22 44Ø A = Ø: GET A$: IF A$ = E$
        THEN POP : RETURN
23 45Ø A = VAL (A$): RETURN
68 46Ø REM  CENTER MESSAGE
CØ 47Ø VTAB V: HTAB ( INT ((4Ø -
        LEN (M$)) / 2) + 1): PRI
       NT M$: RETURN
F6 48Ø REM  DRAW A LINE
66 49Ø VTAB V: FOR I = 1 TD 4: P
       RINT L$;: NEXT : RETURN
29 5ØØ REM  GET DRIVE
84 51Ø PRINT "DRIVE : ";D; CHR$
       (B);
1Ø 52Ø GOSUB 44Ø: IF A$ = CHR$ (
       13) THEN A = 1
82 53Ø IF A < 1 DR A > 2 THEN 52
       Ø
7E 54Ø D = A: RETURN
7D 55Ø REM   GET PAGE
ED 56Ø GOSUB 44Ø: IF A < Ø OR A
       > 2 THEN 56Ø
E4 57Ø P = A: RETURN
ØA 58Ø REM  ASK 'ARE YOU SURE ?
       '
D6 59Ø PRINT "ARE YOU SURE ? Y";
        CHR$ (B);
67 6ØØ GET A$: IF A$ = "N"-OR A$
        = E$ THEN PRINT A$;: POP
        : RETURN
24 61Ø IF A$ = CHR$ (13) OR A$ =
        "Y" THEN RETURN
16 62Ø GOTO 6ØØ
21 63Ø REM  CUSTOM MENU ROUTINE
F1 64Ø V = 2: GOSUB 49Ø
F3 65Ø V = 4: GOSUB 47Ø
76 66Ø V = 6: GOSUB 49Ø
21 67Ø PRINT : VTAB 9
AB 6BØ FOR I = 1 TO X(J): HTAB 2
       : PRINT TI$(J,I): NEXT
3A 69Ø I = 1: VTAB 24: CALL - 86
       B
E7 7ØØ VTAB I + B: HTAB 2: INVER
       SE : PRINT TI$(J,I): NORM
       AL
54 71Ø A = PEEK ( - 163B4): IF A
        < 12B THEN 71Ø
C4 72Ø POKE - 1636B,Ø:A = A - 12
       B
CF 73Ø IF A = 21 OR A = 1Ø THEN
       77Ø
CE 74Ø IF A = B OR A = 11 THEN B
       ØØ
A9 75Ø IF A = 13 THEN RETURN
AØ 76Ø GOTO 71Ø
89 77Ø VTAB I + B: HTAB 2: PRINT
        TI$(J,I)
C8 7BØ IF I + 1 > X(J) THEN I =
       1: GDTD 7ØØ
6D 79Ø I = I + 1: GDTD 7ØØ
7C BØØ VTAB I + B: HTAB 2: PRINT
        TI$(J,I)
AE B1Ø IF I = 1 THEN I = X(J): G
       OTD 7ØØ
31 B2Ø I = I - 1: GDTD 7ØØ
99 B3Ø REM  SCREEN EDITOR
8E B4Ø HOME : TEXT :J = 2:M$ = "
       SCREEN EDITDR": GOSUB 64Ø
EE B5Ø IF I = X(J) THEN RETURN
D3 86Ø ON I GOSUB B9Ø,93Ø,97Ø,1Ø
       2Ø,11ØØ,119Ø,126Ø
27 87Ø GOTD B4Ø
65 BBØ REM   DISPLAY SCREEN
3A B9Ø VTAB 23: PRINT : PRINT "D
       ISPLAY SCREEN : ";
7Ø 9ØØ GOSUB 56Ø: IF A = Ø THEN
       RETURN
9D 91Ø PDKE 255,P: CALL BA: GOTD
        9ØØ
6D 92Ø REM   INVERT SCREEN
86 93Ø VTAB 23: PRINT : PRINT "I
       NVERT SCREEN : ";
78 94Ø GDSUB 56Ø: IF A = Ø THEN
       RETURN
28 95Ø PDKE 255,P: CALL BA + 3:
       RETURN
2Ø 96Ø REM   CDPY SCREEN
14 97Ø VTAB 22: PRINT : PRINT "C
       OPY SCREEN ";: GDSUB 56Ø:
        IF A = Ø THEN RETURN
73 9BØ PDKE 255,P: PRINT P;" TO
       ";3 - P
68 99Ø GDSUB 59Ø
AC 1ØØØ CALL BA + 6: RETURN
C5 1Ø1Ø REM   SUPERIMPOSE SCREEN
14 1Ø2Ø VTAB 21: PRINT : PRINT "
       SUPERIMPOSE SCREEN ";: G
       DSUB 56Ø: IF A = Ø THEN
       RETURN
72 1Ø3Ø POKE 255,P: PRINT P;" TO
        ";3 - P
6C 1Ø4Ø PRINT "1 : ORA  2 : ANO
        3 : EDR   CHDDSE : ";
4A 1Ø5Ø GDSUB 44Ø
7F 1Ø6Ø IF A < 1 OR A > 3 THEN 1
       Ø5Ø
31 1Ø7Ø PRINT A: POKE 254,A: GOS
       UB 59Ø
E4 1ØBØ CALL BA + 9: RETURN
AA 1Ø9Ø REM   COLOR SCREEN
E2 11ØØ VTAB 21: PRINT : PRINT "
       COLOR SCREEN : ";: GOSUB
        56Ø: IF A = Ø THEN RETU
       RN
F1 111Ø PRINT P: PRINT "COLOR :
        ";
A4 112Ø GET A$: IF A$ = E$ THEN
       RETURN
98 113Ø IF A$ = "Ø" THEN C = Ø:
       GOTO 115Ø
ØB 114Ø C = VAL (A$): IF C < 1 O
       R C > 7 THEN 112Ø
72 115Ø PRINT C: GOSUB 59Ø
98 116Ø POKE 23Ø,32 * P: HCOLOR=
        C: HPLOT Ø,Ø: CALL 6245
       4
EF 117Ø RETURN
C9 118Ø REM   FLIP HI BITS
87 119Ø VTAB 21: PRINT : PRINT "
       FLIP HI BITS ON SCREEN :
        ";: GOSUB 56Ø: IF A = Ø
        THEN RETURN
46 12ØØ PRINT P: POKE 255,P: PRI
       NT "1 : SET  2 : CLEAR
       3 : FLIP  CHOOSE : ";
3E 121Ø GOSUB 44Ø
43 122Ø IF A < 1 OR A > 3 THEN 1
       21Ø
34 123Ø PRINT A;: PDKE 254,A: GD
       SUB 59Ø
5C 124Ø CALL BA + 12: RETURN
55 125Ø REM   SWAP SCREENS
3A 126Ø VTAB 23: PRINT : GDSUB 5
       9Ø
98 127Ø CALL BA + 15: RETURN
FD 12BØ REM  LOAO SCREEN
AE 129Ø VTAB 2Ø: PRINT : PRINT "
       LDAO TD SCREEN : ";: GOS
       UB 56Ø: IF A = Ø THEN RE
       TURN
8C 13ØØ PRINT P: GDSUB 51Ø: IF A
        = Ø THEN RETURN
15 131Ø PRINT D: INPUT "FILENAME
        : ";F$
C3 132Ø IF F$ = "" THEN RETURN
35 133Ø VTAB 1: PRINT : PRINT D$
       ;"BLOAD";F$;",D";O;",A";
       P * B192
E7 134Ø RETURN
97 135Ø REM  SAVE SCREEN
2Ø 136Ø VTAB 2Ø: PRINT : PRINT "
       SAVE SCREEN : ";: GDSUB
       56Ø: IF A = Ø THEN RETUR
       N
A8 137Ø PRINT P: GDSUB 51Ø: IF A
        = Ø THEN RETURN
31 13BØ PRINT O: INPUT "FILENAME
        : ";F$
DF 139Ø IF F$ = "" THEN RETURN
11 14ØØ VTAB 1: PRINT : PRINT O$
       ;"BSAVE";F$;",O";D;",A";
       P * B192;",LB192"
DD 141Ø RETURN
F6 142Ø REM  CATALDG DISK
36 143Ø VTAB 23: PRINT : GOSUB 5
       1Ø: IF A = Ø THEN RETURN
22 144Ø HOME :M$ = "CATALOG DF O
       RIVE " + STR$ (D):V = 1:
        GOSUB 47Ø
95 145Ø V = 2: GOSUB 49Ø
CA 146Ø POKE 34,2: PRINT : PRINT
        D$"CATALDG,D"D
91 147Ø V = 2:M$ = " PRESS A KEY
       ...  ": GOSUB 47Ø
38 148Ø VTAB 2: HTAB 27: GET T$:
        PDKE 34,Ø: RETURN       ©
```

COMMODORE 64 Key Phantom

Melvin Baker

By expanding the 64's internal keyboard buffer, you can use the dynamic keyboard technique for very powerful effects. This machine language utility does all the hard work for you, even if you know nothing about machine language. A disk drive is required.

If you've been following Jim Butterfield's recent series on dynamic keyboard programming (COMPUTE!, October–December 1985), you know that this technique is a powerful programming tool. By making the computer "type on its own keyboard," you can write programs that modify themselves as they run, enter direct mode commands, and do many other things that ordinarily are difficult or impossible from within a program.

The dynamic keyboard technique works by POKEing the desired character codes into an area of memory called the keyboard buffer, which normally starts at location 631. This is where the computer receives keystrokes, so POKEing character codes into the buffer makes the computer think those keys have been pressed. Next, you POKE the number of characters in the buffer into the keyboard buffer counter at location 198. When the program ends, the computer types the codes in the buffer, just as if you pressed the same keys yourself.

However, the dynamic keyboard technique suffers from one major limitation. Since the keyboard buffer can't hold more than ten characters, you're limited to fairly short commands. If your command takes more than ten characters to type (including a carriage return), it simply won't fit into the buffer.

"Commodore 64 Key Phantom" overcomes this limitation by relocating and expanding the 64's keyboard buffer in a free memory area. When the machine language (ML) portion of Key Phantom is active, the 64 has a keyboard buffer 3,758 characters in length—enough to permit very elaborate command sequences.

A Phantom Typist

Before we get into the details of how Key Phantom works, let's try a short demonstration. Type in and save the program listed below. When you run it, the program automatically POKEs the ML code into memory and then displays a three-option menu on the screen. By pressing a number key from 1–3 you can create a new commands file, execute an existing commands file, or exit the program.

To get started, press 1 to create a new commands file. This file will be called COMMANDS on the disk, so if your disk already contains a sequential file of that name, you should exit the program and copy the old file to another disk before proceeding.

Option 1 is a simple text editor which lets you store a series of character codes in the file named COMMANDS. Later on, the Key Phantom can read the character codes from this file and type them with the dynamic keyboard technique. When you choose Option 1, the screen clears and displays a message indicating which line of the commands file is being edited. The line number is solely for your information—it won't become part of the file. Type in the following lines exactly as shown. Where you see the name of a key enclosed in curly braces { } you should press the key indicated inside the braces. For example, press RETURN when you see {RETURN}. Press the cursor-down key when you see {DOWN}.

```
Key£01 Phantom£02 Demonstration£03
  {DOWN}{RETURN}
Watch me type in a line£01{DOWN}
  {RETURN}
that changes the screen colors...£02
  {DOWN}{RETURN}
f£01o£01r£01j=1£02to£02200:£02poke
  £0253280,£02j:£02n£01e£01x£01t£03
  {RETURN}
```

Use the DEL key to erase any mistakes within a line. When all four lines have been entered, press any key at the prompt to write the commands file to disk. At this point, the program returns you to the main menu. Press the 2 key to execute the commands file. After a brief pause while the ML code is placed in memory, the program loads the commands file.

Now the Key Phantom begins typing the characters from your commands file directly on the screen. Although the READY prompt and blinking cursor appear on the screen, you are not in BASIC ready mode. The Key Phantom has control of the computer until it reaches the end of the commands file. You should see the following display:

```
Key Phantom Demonstration
Watch me type in a line
that changes the screen colors...
forj=1to200:poke53280,j:next
```

Note the time delays of various durations that are used at different points in the printing sequence. These result from the characters £01, £02, and £03 that you typed when creating the file. The £ character tells the Phantom to pause the printing for the number of seconds specified in the following number.

The delay number must be expressed in hexadecimal (base 16). Thus, £01 pauses the printing for one second; £0F pauses for 15 seconds, and so on. By including delays in the character sequence, you can print information at any speed you like.

Pseudo-Keys

The £ character is an example of a Key Phantom *pseudo-key.* Instead of printing something on the screen, a pseudo-key performs a certain action. A second pseudo-key is the back-arrow key, located at the upper-left corner of the keyboard. When you include this character in a command sequence, Key Phantom waits for you to type a line from the keyboard. The input terminates when you press RETURN, just like INPUT in a BASIC program.

Since Key Phantom essentially types every character from the command file, you must keep in mind what would happen if you were typing those characters yourself. For instance, it's necessary to print a cursor down character before printing RETURN at the ends of the first three example lines. Otherwise you'd get a SYNTAX ERROR, since those lines don't contain BASIC commands. But no cursor down is used at the end of the last line: In this case you want to press RETURN at the end of the line to make the computer perform those actions.

Cursor control characters can be used for a variety of different effects. Just as in BASIC, you can move left, right, up, or down, go to the home position, clear the screen, and so on. The editor accepts any keys except DEL, CTRL, RUN/STOP, and RESTORE. Since control characters would garble the editor's screen display, it generally displays a reverse video < or > symbol to show that a control character was typed. If you need to perform an action not available from the keyboard, you can always execute a short PRINT statement. For instance, PRINT CHR$(14)CHR$ (8) locks the keyboard into lowercase/uppercase mode.

Advanced Applications

Because the ML portion of Key Phantom is driven by the computer's hardware interrupt routine, it can operate while a BASIC program is running. This means you can use it to feed input directly to a running BASIC program.

When would this be useful? To take a simple example, let's say you use a certain BASIC program frequently: It could be a checkbook program, an events calendar, or whatever. The program may begin by asking you to choose from several different options, input various items of information, and so on. By writing an appropriate command file for Key Phantom, you could make the computer load and run the BASIC program, select the option or options you want, and input as many items of information as needed. If you need to input new information at any point, the back arrow pseudo-key lets you do so. And when automatic control is no longer needed, the command file can terminate, leaving you in the BASIC program as usual.

If you're the type who likes to have several programming aids active at once, why not write a Key Phantom command file that automatically loads and activates all your favorite utilities at once? When you begin using Key Phantom, you'll probably think of many more uses as well.

Of course, since it uses memory from locations 49152–53247, this program is not compatible with utilities that use the same memory area. And you must be careful not to activate any other ML routines that disturb the 64's hardware interrupt vector at locations 788–789 ($0314–$0315). You should also look out for BASIC programs that begin by clearing the computer's keyboard buffer—to avoid losing information from the command file, you should pause Key Phantom for a few seconds (with the £ pseudo-key) when the program begins.

When feeding input to a program, you must keep in mind what sort of input the program expects. If the program accepts data with INPUT, you should terminate the corresponding data item with a carriage return. On the other hand, if the program accepts data with GET, you should not end the data with a carriage return. GET usually takes a single keypress, not a keypress plus a carriage return.

Occasionally you may find a program that needs to use Key Phantom pseudo-keys for its own purposes. Then you'll have to change the pseudo-keys to some other character. This can be done by substituting different character codes in lines 570 and 590. The REMs in the program indicate which value belongs to which pseudo-key.

Commodore 64 Key Phantom

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" published in this issue of COMPUTE!

```
JH 100 ZZ=53368
HJ 110 CLOSE15:PRINT"{CLR}
       {2 DOWN}KEY PHANTOM
       {2 DOWN}"
GQ 120 PRINT"1] EDITOR{DOWN}":
       PRINT"2] EXECUTE{DOWN}"
       :PRINT"3] EXIT{DOWN}"
CH 130 GOSUB820:K=VAL(Q$):IFK<
       1ORK>3THEN130
AP 140 ON K GOTO150,460,510
FB 150 OPEN15,8,15,"I"
XG 160 GOSUB520:PRINT:PRINTD$
FK 170 OPEN5,8,5,"0:COMMANDS,S
       ,W":GOSUB520
HR 180 PRINT:PRINTD$:IF A1<20T
       HEN260
BK 190 CLOSE5
QG 200 IP A1<>63THEN110
JQ 210 PRINT"{DOWN}1] SCRATCH
       {DOWN}":PRINT"2] APPEND
       {DOWN}":PRINT"3] MENU
       {DOWN}"
RQ 220 GOSUB820:K=VAL(Q$):IPK<
       1ORK>3THEN220
CF 230 ON K GOTQ240,250,110
XD 240 PRINT#15,"S:COMMANDS":G
       OTO160
MK 250 OPEN5,8,5,"0:COMMANDS,S
       ,A":GOSUB520:GOTO180
QF 260 FOR LN=1TO4000:NEXT:LN=
       1
PG 270 PRINT"{CLR}{RIGHT}KEY P
       HANTOM "D$"{DOWN}":PRIN
       T"LINE - "LN"{DOWN}":L$
       =""
HC 280 PRINT"{RVS} {OFF}{LEFT}
       ";
MP 290 GOSUB820:K=A8C(Q$):IP K
       <32THENQ$="{RVS}<{OFF}"
XQ 300 IF K>127AND K<160THENQ$
       ="{RVS}>{OFP}"
HA 310 IF K=34THENQ$="{RVS}'
       {OFP}"
CA 320 IF K=20THENPRINT"
       {2 LEFT}";:K=LEN(L$)-1:
       L$=LEFT$(L$,K-(K<0/-2))
       :GOTO280
RK 330 PRINTQ$;
AQ 340 IF K<>13THENL$=L$+CHR$(
       K):GOTO280
DF 350 PRINT:PRINT"{DOWN}[RET]
        FOR NEXT LINE{DOWN}"
JC 360 PRINT"[DEL] TO REDO LIN
       E{DOWN}"
SR 370 PRINT"ANY OTHER TO EXIT
       {DOWN}"
GP 380 GOSUB820
```

```
DH 39Ø IFQ$=CHR$(13)THENLN=LN+
       1:PRINT#5,L$:GOSUB52Ø:G
       OTO27Ø
FQ 4ØØ IFQ$=CHR$(2Ø)THEN27Ø
DQ 41Ø PRINT:PRINT"ARE YOU SUR
       E [Y/N]? ";
GB 42Ø GOSUB82Ø:IFQ$<>"Y"ANDQ$
       <>"N"THEN42Ø
JS 43Ø PRINTQ$:IFQ$="N"THEN35Ø
QF 44Ø PRINT#5,L$:GOSUB52Ø:PRI
       NT:PRINTD$
HE 45Ø CLOSE5:GOSUB52Ø:PRINT:P
       RINTD$:PRINT:CLOSE15:GO
       TO11Ø
XR 46Ø RESTORE:CS=Ø:READ LB,HB
       :A1=HB*256+LB:A2=A1:PRI
       NT"LOADING AT"A1;
MC 47Ø READK:PRINT" >{LEFT}";:
       IFK<ØTHEN49Ø
QH 48Ø CS=CS+K:POKEA2,K:A2=A2+
       1:GOTO47Ø
KC 49Ø IF CS<>ZZ THENPRINT"
       {RVS} CHECKSUM ERROR
       {OFF}";CS:GOTO11Ø
JG 5ØØ SYS A1
FH 51Ø PRINT"{CLR}":END
PQ 52Ø INPUT#15,A1,D$,A2,A3
EJ 53Ø IF A1<2ØTHENRETURN
CB 54Ø D$="{RVS} "+D$+" {OFF}"
       :RETURN
RH 55Ø DATA Ø,192,76,81,193,17
       3,183,192,2Ø1,Ø,2Ø8,79,
       173,185,192,2Ø1,Ø
QF 56Ø DATA 2Ø8,34,173,198,Ø,2
       Ø1,Ø,2Ø8,24,32,154,192,
       2Ø1,Ø,24Ø,41,2Ø1
GE 57Ø DATA 92:REM COMMAND ONE
        (£)
AP 58Ø DATA 24Ø,75,2Ø1
PR 59Ø DATA 95:REM COMMAND TWO
        (←)
GA 6ØØ DATA 24Ø,27,141,119,2,2
       38,198,Ø,32,158,192,76
DM 61Ø DATA 226,252,173,197,Ø,
       2Ø1,1,2Ø8,246,169,Ø,141
       ,185,192,76,48,192
RP 62Ø DATA 238,185,192,76,45,
       192,12Ø,173,49,192,141,
       2Ø,3,173,5Ø,192,141
FC 63Ø DATA 21,3,88,76,48,192,
       238,184,192,173,184,192
       ,2Ø1,6Ø,2Ø8,8,169
EA 64Ø DATA Ø,141,184,192,2Ø6,
       183,192,76,48,192,32,13
       2,192,1Ø,1Ø,1Ø,1Ø
PB 65Ø DATA 141,183,192,32,132
       ,192,13,183,192,141,183
       ,192,76,45,192,32,158
BB 66Ø DATA 192,32,154,192,162
       ,Ø,221,186,192,24Ø,7,23
       2,224,16,2Ø8,246,162
CR 67Ø DATA Ø,138,96,173,81,19
       3,96,238,155,192,2Ø8,5,
       238,156,192,24Ø,16Ø
BC 68Ø DATA 173,156,192,2Ø1,2Ø
       7,2Ø8,7,173,155,192,2Ø1
       ,255,24Ø,146,96,Ø.Ø
MF 69Ø DATA Ø,48,49,5Ø,51,52,5
       3,54,55,56,57,65,66,67,
       68,69,7Ø
FK 7ØØ DATA 48,58,67,79,77,77,
       65,78,68,83,32,2Ø7,255,
       176,74,145,253
CM 71Ø DATA 23Ø,253,2Ø8,2,23Ø,
       254,165,254,2Ø1,2Ø7,2Ø8
       ,6,165,253,2Ø1,255,24Ø
FM 72Ø DATA 54,32,183,255,41,6
       4,24Ø,224,169,Ø,145,253
       ,169,5,32,195,255
BM 73Ø DATA 32,51,193,169,15,3
       2,195,255,176,28,32,2Ø4
       ,255,12Ø,173,2Ø,3
CF 74Ø DATA 141,49,192,173,21,
       3,141,5Ø,192,169,3,141,
       2Ø,3,169,192,141
BK 75Ø DATA 21,3,88,96,169,5,3
       2,195,255,169,15,32,195
       ,255,32,138,255
SS 76Ø DATA 76,131,164,176,238
       ,162,15,32,198,255,176,
       231,32,2Ø7,255,176,226
SB 77Ø DATA 72,32,2Ø7,255,176,
       22Ø,2Ø1,13,2Ø8,247,1Ø4,
       2Ø1,48,2Ø8,211,96,169
GR 78Ø DATA 15,162,8,16Ø,15,32
       ,186,255,169,Ø,32,189,2
       55,32,192,255,32
HH 79Ø DATA 51,193,169,5,162,8
       ,16Ø,5,32,186,255,169,1
       Ø,162,2Ø2,16Ø,192
JR 8ØØ DATA 32,189,255,32,192,
       255,32,51,193,162,5,32,
       198,255,176,159,169
ES 81Ø DATA 81,133,253,169,193
       ,133,254,16Ø,Ø,76,212,1
       92,255,-1
KG 82Ø GETQ$:IFQ$=""THEN82Ø
GP 83Ø RETURN
```

©

PRINTERS

EPSON	
LX90	$234 95
HS80 Letterjet	318 95
JX80	448 95
FX185	454 95
LQ1500 (PAR)	949 95
FX85	331 95
LX80	210 95
STAR MICRONICS	
SG10	$210 95
SG10C	234 95
SG15	368 95
SD10	321 95
SD15	440 95
SR10	468 95
SR15	581 95
Power type	299 95
CITIZEN	
MSP10	$274 95
MSP15	438 95
MSP20	438 95
MSP25	548 95
PANASONIC	
1091	$231 95
1092	368 95
1093	424 95
3151	424 95
3131	259 95
OKIDATA	
Okimate 10	$170 95
Okimate 20	209 95
182	219 95
192	349 95
193	514 95
84	640 95
LEGEND	
808/NLQ	$149 95
1080 NLQ	205 95
1380	259 95
13805	294 95
JUKI	
6000	$189 95
6100	379 95
6300	769 95
TOSHIBA	
P351	$1289 95
1340	599 95
DAISYWRITER	
2000	$739 95

N.E.C.	
2030	$699 95
2050	699 95
3530	1309 95
3550	1389 95

MODEMS

ANCHOR	
Volkmodem 12	$179 95
Signalman Express	299 95
Lightning 2400	399 95
Westridge	59 95
DIGITAL DEVICES	
300 Baud (Atari)	$98 95
HAYES	
300	$137 95
300/1200	386 95
1200 B w/software	378 95
1200B	349 95
NOVATION	
Smart Cat Plus (1200)	$309 95
Professional 2400	629 95
SUPRA	
MPP1000E (Atari)	$54 95
MPP1064(C-64)	54 95
CAL-ABCO	
Smart Team 1200	$199.95
C.D.I.	
Mitey Mo (64)	$59 95

INTERFACES

SUPRA	
MTP1150	$45 95
DIGITAL DEVICES	
U Print/Port	$49 95
U Print/16K	69 95
U Print/64K	79 95
CARDCO	
G-Wiz	$48 95
AT-1 (Atari)	48 95
TYMAC	
Epson	$58 95
PAR (Centronics Std)	54 95

DISKETTES

BONUS	5¼	3½
SS/DD	$ 9.50	—
DS/DD	13.50	—
NO LABEL (with Pen & Flip-n-file)		
SS/DD	$10.50	—
DS/DD	14.50	—
MAXELL		
MD1	$15.95	32.95
MD2	19.95	38.95
MEMOREX		
SS/DD	$12.50	29.95
DS/DD	16.50	34.95
FF30/20SS/DD	—	59.95
FF50/20SS/DD	36.95	—
FF50/20DS/DD	42.95	—

All Diskettes Carry a Lifetime Warranty

MONITORS

PRINCETON	
Max 12E	$189 95
HX 9	459 95
HX-9E	519 95
HX 12	469 95
HX 12E	555 95
SR-12	589 95
SR-12E	649 95
NEC	
1201 (Green)	$99 95
1205 (Amber)	99 95
1280 (TTL Green)	148 95
1285 (TTL Amber)	148 95
1460	269 95
1401 (R G B)	659 95
ZENITH	
ZVM 122/123	$ 74 95
ZVM 124 (TTL Amber)	149 95
ZVM 131 (RGB)	279 95
ZVM 135 (RGB/Color)	459 95
TEKNIKA	
MJ-10	$178 95
MJ-22	254 95
AMDEK	
300G	$116 95
300A	126 95
310A	144 95
Color 300	175 95
Color 500	339 95
Color 600	398 95
Color 700	468 95
Color 710	538 95
Color 722	579 95

BOARDS

AST	
Six Pack Plus	$239 95
Advantage (AT)	399 95
Graph Pak	589 95
Preview	299 95
Mono Graph Plus	389 95
5251/11	799 95
5251/12	579 95
EVEREX	
Color Card	$295 95
Magic Card	165 95
HERCULES	
Color	$169 95
Graphics	289 95
PARADISE	
Five Pack	$159 95
Mono	149 95
Modular Graphics	279 95
Multi Display	209 95
TECMAR	
Graphics Master	$459 95
Captain 64	$189 95

DISK DRIVES

INDUS	
GT (Atari)	$198.95
GT (Commodore)	218 95
TANDON	
TM100-2	$109.95
TEAC	
55B	$104 95

WESTERN REGION
1-800-351-3455
In CA Call: 1-800-351-3422

ORDER TOLL FREE

EASTERN REGION
1-800-351-3442

"Where Prices are Born, Not Raised."

WHITE HOUSE COMPUTER

WESTERN
REGION
11327 Trade Center Drive
Suite 335
Rancho Cordova, CA 95670
Customer Service: 916-635-3455

No deposit on C O D orders Free freight on all prepaid cash orders over $300 in the Continental U S A APO and FPO orders add $5.00 per hundred For Priority Mail add $10.00 per hundred Free shipping for PA residents PA residents add 6% sales tax All defective products must have a prior RA number

Hours Monday thru Friday 9 a m 6 p m

EASTERN
REGION
P.O. Box 4025
Williamsport, PA 17701
Customer Service:
717-322-7700

MASTER CARD 4% VISA 4% AMERICAN EXPRESS 5%

Screen Clock For IBM

Marc Sugiyama

Have you ever become submerged in a project while working on your computer and suddenly discovered it is hours past your bedtime? Or maybe you need to keep a detailed log of your worktime on the computer for business or tax purposes. If so, this utility is the answer—it constantly displays all this information and more on your monitor screen. It works with IBM PC and PCjr computers using DOS 2.0 or higher.

Large mainframe computers generally provide a *sysline* on the terminal screen which tells you the current date and time, who has logged on or off, and whether you've received any new electronic mail. Obviously, not all of these things apply to single-user personal computers, but some of the features would be nice to have.

"Screen Clock" is a short machine language program that prints the day of the week, date, current time, and log-on time at the top of the screen. This information appears no matter what else your computer is doing. You can be running a word processor, copying files, programming, or whatever—the day, date, and time will always be visible.

You might be wondering how it's possible to keep Screen Clock active while running another program; an IBM PC with PC-DOS isn't capable of multitasking. Screen Clock gets around this restriction by not using any PC-DOS function calls, relying instead on the BIOS (Basic Input/Output System) to handle the screen. This has several fortunate consequences:

• Sysline updates are not redirected to a file if you're using DOS file redirection.

• Sysline updates are not printed if you're echoing output to the printer. (But the sysline *is* printed if you press PrtSc for a screen dump.)

• Screen Clock always updates the current "active" screen. It doesn't matter if you switch from the monochrome monitor to the color monitor, change pages in the color screens, or even enter a graphics mode—the date and time are always there.

Winding Up The Clock

Type in the program listing below, save a copy on disk, then type RUN. The program is a BASIC loader that creates a machine language file on your disk with the filename CLOCK.COM. To start the clock, simply type CLOCK (upper- or lowercase is fine) at the A> DOS prompt. A sysline similar to this should appear on the top line of your screen:

```
Wed Jan 01, 1986 12:01A (00:37)
```

The day of the week, date, and current time are self-explanatory. The figure in parentheses is the elapsed time (in hours and minutes) since Screen Clock was started or reset. This "log-on" time runs up to 23 hours and 59 minutes, then rolls over to 00:00.

When you run Screen Clock from DOS, you can select various options by appending commands after typing CLOCK. Each command consists of a slash (/) symbol, a character, and sometimes a number. Here are the commands and options:

/C*n* (Chime) where *n* is an integer from 0 to 3. /C0 means no chiming; /C1 makes the clock chime hourly; /C2 chimes every half-hour; and /C3 chimes every 15 minutes. A chime is a low beep which lasts for less than one second. Even if the screen updates are turned off, Screen Clock always chimes if you have told it to. The default is no chiming.

/U*n* (Update) where *n* is an integer from 1 to 9. This sets how often screen updates are to take place—*n* is the number of half-seconds between updates. The more frequent the updates, the more often the date and time are refreshed on the screen. However, more frequent updates also make other programs run more slowly. The default is equivalent to /U2 (one second between updates).

/M (Military time). This selects military (24-hour) time.

/S (Standard time). This selects standard 12-hour time with an a.m./p.m. marker. Screen Clock defaults to standard time.

/R (Reset). This resets the log-on timer. Screen Clock automatically resets itself to 00:00 when first run.

For example, typing CLOCK /U3/M/C1 at the DOS prompt loads and runs Screen Clock, sets updates every 1½ seconds, sets military time, and makes the clock chime every hour.

The Disappearing Clock

Occasionally, the Screen Clock sysline may get in the way. For example, it may hide text printed on the top line of the screen. You can make it disappear by pressing CTRL and both SHIFT keys simultaneously. Pressing this combination again turns the sysline back on.

Since Screen Clock maintains its own clock, it might not agree precisely with the DOS clock. Generally, it's never more than half a minute off.

Note that the day, date, and

time are reset every time you run Screen Clock. If you change the system date and time, you can reset Screen Clock by running it again. For example, the following would reset the display to 8:00 p.m. on February 14 (the A> prompts are supplied by DOS):

```
A> time 20:00:00
A> date 02-15-86
A> clock
```

The log-on time is not reset unless you append the /R command to CLOCK.

Although Screen Clock makes it appear that your computer is doing more than one thing at a time, it's important to remember that computers can really perform only *one* task at a time (a factor of the basic architecture of all personal computers to date). If the computer spends some if its time updating the sysline, that's time away from running the main program. Thus, the more often the sysline is updated, the more time it steals from the computer, and the slower the main program seems to run. However, the part of Screen Clock that takes the most time is printing the sysline on the screen. If screen updates are turned off, there is virtually no slowdown. So during heavy number crunching you might want to turn the sysline updates off.

I've been using Screen Clock quite a bit and haven't noticed much loss of performance at all. It seems that the computer spends a lot of its time waiting for input (from the keyboard, the disk drives, and so on); all we're doing is giving it something else to do in its "spare time." I have yet to find a program which doesn't work with Screen Clock.

As the power of personal computers increases, it becomes possible to include features once found only on large mainframe computers. A sysline such as Screen Clock is another step in this direction.

How It Works

Mainframe syslines are generally on the bottom row of the screen. The Screen Clock sysline, however, must be on the top row because there's no way via PC-DOS to keep the bottom row from scrolling. The sysline would keep traveling up the screen every time the screen was scrolled. By placing the sysline on the top row, it can be refreshed each time it scrolls off the top of the screen.

The program itself is broken into two sections, resident and nonresident. The resident portion updates the internal counters, sounds the chimes, and updates the screen display. It's driven by the user interrupt 1Ch and is executed about 18 times a second. The nonresident part sets the initial date and time and changes the program's options.

When you execute CLOCK.COM, the program first checks to see if the resident portion is already installed. This is important only when the program returns control to DOS. Then it sets the current date and time and checks for any optional parameters. After this, the program is ready to return to DOS. If the program was already installed, it simply returns to DOS and does nothing else. If it needs to be installed, it first deallocates the environment space, then returns to DOS with the "terminate but stay resident" call to store the resident portion of the program safely in memory.

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

IBM Screen Clock

```
EF 1ØØ CLS:LOCATE 1Ø,1Ø:PRINT"Wr
       iting file ..."
DM 11Ø OPEN "clock.com" FOR OUTP
       UT AS #1
EL 12Ø FOR I=1 TO 131Ø:READ BYTE
       :CKSUM=CKSUM+BYTE:IF BYTE
       <Ø THEN FOR J=1 TO ABS(BY
       TE):PRINT#1,CHR$(Ø);:NEXT
        J:GOTO 14Ø
BH 13Ø PRINT#1,CHR$(BYTE);
GO 14Ø NEXT I:CLOSE 1
FJ 15Ø IF CKSUM <> 124185 THEN P
       RINT"** Error in DATA sta
       tements **":KILL "clock.c
       om":STOP
HJ 16Ø PRINT:PRINT"File for cloc
       k.com has been created.":
       ENO
KN  2ØØ OATA 233,51,4,74,97,11Ø,
       32,7Ø,1Ø1,98,32,77
DG  21Ø OATA 97,114,32,65,112,11
       4,32,77,97,121,32,74
EB  22Ø DATA 117,11Ø,32,74,117,1
       Ø8,32,65,117,1Ø3,32,83
NP  23Ø DATA 1Ø1,112,32,79,99,11
       6,32,78,111,118,32,68
NJ  24Ø OATA 1Ø1,99,32,31,28,31,
       3Ø,31,3Ø,31,31,3Ø
FP  25Ø OATA 31,3Ø,31,83,117,11Ø
       ,32,77,111,11Ø,32,84
JK  26Ø DATA 117,1Ø1,32,87,1Ø1,1
       ØØ,32,84,1Ø4,117,32,7Ø
PH  27Ø OATA 114,1Ø5,32,83,97,11
       6,32,-6,1,Ø,1,8Ø
AE  28Ø DATA 19,2,-5,24Ø,18,Ø,1,
       -86,13,255,8Ø,97
NL  29Ø DATA 117,1Ø8,8Ø,83,81,82
       ,86,87,85,3Ø,6,14Ø
GK  3ØØ DATA 2ØØ,142,216,142,192
       ,232,198,1,232,45,Ø,232
DP  31Ø DATA 133,Ø,16Ø,1Ø8,1,58,
       6,1Ø9,1,114,23,187
BM  32Ø DATA 91,1,232,158,1,137,
       14,95,1,232,144,Ø
CJ  33Ø DATA 128,62,111,1,Ø,116,
       3,232,25Ø,Ø,7,31
GI  34Ø DATA 93,95,94,9Ø,89,91,8
       8,2Ø7,18Ø,2,2Ø5,22
NC  35Ø DATA 36,7,6Ø,7,116,6,198
       ,6,115,1,Ø,195
DN  36Ø DATA 128,62,115,1,Ø,117,
       67,128,54,111,1,1
BJ  37Ø DATA 198,6,115,1,1,128,6
       2,111,1,Ø,116,4
PE  38Ø DATA 232,197,Ø,195,18Ø,1
       5,2Ø5,16,136,62,114,1
CN  39Ø DATA 18Ø,3,2Ø5,16,137,22
       ,112,1,18Ø,2,186,-2
DH  4ØØ DATA 2Ø5,16,185,31,Ø,176
       ,32,18Ø,14,2Ø5,16,226
HO  41Ø DATA 25Ø,18Ø,2,138,62,11
       4,1,139,22,112,1,2Ø5
PM  42Ø DATA 16,195,128,62,199,1
       ,255,116,25,16Ø,199,1
NA  43Ø DATA 58,6,198,1,119,5,25
       4,6,199,1,195,198
JA  44Ø DATA 6,199,1,255,228,97,
       36,252,23Ø,97,195,138
EA  45Ø DATA 22,197,1,128,25Ø,Ø,
       117,1,195,128,62,95
DO  46Ø DATA 1,Ø,117,8,128,62,19
       6,1,Ø,116,62,195
DC  47Ø DATA 128,25Ø,1,116,5Ø,12
       8,62,95,1,3Ø,117,8
PN  48Ø DATA 128,62,196,1,Ø,116,
       42,195,128,25Ø,2,116
KK  49Ø OATA 3Ø,128,62,95,1,15,1
       17,8,128,62,196,1
BF  5ØØ DATA Ø,116,22,195,128,62
       ,95,1,45,117,8,128
ON  51Ø DATA 62,196,1,Ø,116,7,19
       5,198,6,196,1,Ø
PD  52Ø DATA 195,198,6,196,1,1,1
       98,6,199,1,Ø,176
JJ  53Ø DATA 182,23Ø,67,184,1Ø2,
       1Ø,23Ø,66,138,196,23Ø,66
GF  54Ø DATA 228,97,12,3,23Ø,97,
       195,198,6,1Ø8,1,Ø
DI  55Ø OATA 191,116,1,252,139,5
       4,1Ø2,1,2Ø9,23Ø,2Ø9,23Ø
PJ  56Ø OATA 129,198,63,1,185,4,
       Ø,243,164,139,54,97
GC  57Ø DATA 1,2Ø9,23Ø,2Ø9,23Ø,1
       29,198,255,Ø,185,4,Ø
OB  58Ø OATA 243,164,16Ø,99,1,23
       2,22,1,184,44,32,171
FH  59Ø DATA 16Ø,1Ø1,1,232,12,1,
       16Ø,1ØØ,1,232,6,1
FO  6ØØ OATA 176,32,17Ø,139,14,9
       5,1,138,38,11Ø,1,232
CE  61Ø OATA 1,1,184,32,4Ø,171,1
       87,1Ø4,1,232,58,Ø
IP  62Ø DATA 18Ø,1,232,242,Ø,176
       ,41,17Ø,18Ø,15,2Ø5,16
NO  63Ø OATA 136,62,114,1,18Ø,3,
       2Ø5,16,137,22,112,1
FP  64Ø OATA 18Ø,2,186,-2,2Ø5,16
       ,19Ø,116,1,139,2Ø7,43
KH  65Ø OATA 2Ø6,172,18Ø,14,2Ø5,
       16,226,249,18Ø,2,138,62
IH  66Ø DATA 114,1,139,22,112,1,
       2Ø5,16,195,139,87,2
NB  67Ø DATA 139,7,187,69,4,247,
       243,179,6Ø,246,243,138
CK  68Ø OATA 232,138,2Ø4,195,187
       ,91,1,232,16,Ø,115,3
```

```
AO  690 DATA 232,43,Ø,187,1Ø4,1,
        232,5,Ø,254,6,1Ø8
JL  7ØØ DATA 1,195,255,7,117,3,2
        55,71,2,131,127,2
BI  71Ø DATA 24,114,17,129,63,17
        6,Ø,114,11,199,7,-2
FJ  72Ø DATA 199,71,2,-2,249,195
        ,248,195,255,6,1Ø2,1
AH  73Ø DATA 131,62,1Ø2,1,6,118,
        6,199,6,1Ø2,1,-2
iA  74Ø DATA 254,6,99,1,139,22,9
        7,1,232,49,Ø,58
DM  75Ø DATA 22,99,1,115,42,198,
        6,99,1,1,255,6
MI  76Ø DATA 97,1,131,62,97,1,12
        ,118,26,199,6,97
PL  77Ø DATA 1,1,Ø,254,6,1ØØ,1,1
        28,62,1ØØ,1,99
FA  78Ø DATA 118,9,198,6,1ØØ,1,Ø
        ,254,6,1Ø1,1,195
PO  79Ø DATA 138,218,5Ø,255,138,
        151,5Ø,1,128,251,2,117
CA  8ØØ DATA 16,246,6,1ØØ,1,3,11
        7,9,128,62,1ØØ,1
JH  81Ø DATA Ø,116,2,254,194,195
        ,212,1Ø,5,48,48,134
IG  82Ø DATA 196,171,195,182,32,
        128,252,1,116,18,182,65
EB  83Ø DATA 128,253,12,114,5,18
        2,8Ø,128,237,12,1Ø,237
EK  84Ø DATA 117,2,181,12,138,19
        7,232,217,255,176,58,17Ø
JL  85Ø DATA 138,193,232,2Ø9,255
        ,128,254,32,116,3,138,198
MB  86Ø DATA 17Ø,195,82,1Ø1,113,
        117,1Ø5,114,1Ø1,115,32,68
BM  87Ø DATA 79,83,32,5Ø,46,48,3
        2,111,114,32,97,98
CE  88Ø DATA 111,118,1Ø1,46,13,1
        Ø,36,78,111,119,32,1Ø5
PG  89Ø DATA 11Ø,115,116,97,1Ø8,
        1Ø8,1Ø5,11Ø,1Ø3,32,114,1Ø
        1
FI  9ØØ DATA 115,1Ø5,1ØØ,1Ø1,11Ø
        ,116,32,112,111,114,116,1
        Ø5
BH  91Ø DATA 111,11Ø,32,111,1Ø2,
        32,67,76,79,67,75,46
OK  92Ø DATA 13,1Ø,36,39,32,117,
        11Ø,1Ø7,11Ø,111,119,11Ø
CH  93Ø DATA 32,112,97,114,97,1Ø
        9,1Ø1,116,1Ø1,114,46,13
PH  94Ø DATA 1Ø,36,83,112,1Ø1,99
        ,1Ø5,1Ø2,121,32,97,32
EO  95Ø DATA 11Ø,117,1Ø9,98,1Ø1,
        114,32,1Ø2,114,111,1Ø9,32
JK  96Ø DATA 49,45,57,32,1Ø2,111
        ,114,32,39,85,39,32
KG  97Ø DATA 115,119,1Ø5,116,99,
        1Ø4,13,1Ø,36,83,112,1Ø1
JE  98Ø DATA 99,1Ø5,1Ø2,121,32,9
        7,32,11Ø,117,1Ø9,98,1Ø1
FO  99Ø DATA 114,32,1Ø2,114,111,
        1Ø9,32,48,45,51,32,1Ø2
OP  1ØØØ DATA 111,114,32,39,67,3
         9,32,115,119,1Ø5,116,99
JG  1Ø1Ø DATA 1Ø4,13,1Ø,36,-2,47
         ,18Ø,48,2Ø5,33,6Ø,Ø
KI  1Ø2Ø DATA 117,9,186,125,4,18
         Ø,9,2Ø5,33,2Ø5,32,184
CC  1Ø3Ø DATA Ø,55,2Ø5,33,136,22
         ,53,5,187,125,4,177
PF  1Ø4Ø DATA 4,211,235,67,137,3
         Ø,51,5,184,28,53,2Ø5
HM  1Ø5Ø DATA 33,19Ø,2ØØ,1,141,1
         27,252,185,4,Ø,252,243
IE  1Ø6Ø DATA 166,131,249,Ø,116,
         41,18Ø,9,186,154,4,2Ø5
ID  1Ø7Ø DATA 33,184,28,37,186,2
         Ø4,1,2Ø5,33,3Ø,7,232
JE  1Ø8Ø DATA 32,Ø,232,87,Ø,161,
         44,Ø,142,192,18Ø,73
KP  1Ø9Ø DATA 2Ø5,33,184,Ø,49,13
         9,22,51,5,2Ø5,33,232
NJ  11ØØ DATA 8,Ø,232,63,Ø,184,Ø
         ,76,2Ø5,33,6,31
KK  111Ø DATA 18Ø,Ø,2Ø5,26,137,2
         2,91,1,137,14,93,1
BO  112Ø DATA 18Ø,42,2Ø5,33,5Ø,2
         28,163,1Ø2,1,138,198,163
JO  113Ø DATA 97,1,136,22,99,1,1
         98,6,1Ø1,1,19,129
BI  114Ø DATA 233,1Ø8,7,128,249,
         99,118,7,128,233,1ØØ,254
DH  115Ø DATA 6,1Ø1,1,136,14,1ØØ
         ,1,195,3Ø,14,31,19Ø
BF  116Ø DATA 129,Ø,252,172,6Ø,3
         2,116,251,6Ø,13,116,51
BN  117Ø DATA 58,6,53,5,116,241,
         138,224,36,223,6Ø,82
IL  118Ø DATA 116,39,6Ø,77,116,5
         1,6Ø,83,116,55,6Ø,85
DG  119Ø DATA 116,59,6Ø,67,116,9
         Ø,8Ø,178,39,18Ø,2,2Ø5
LP  12ØØ DATA 33,88,138,212,18Ø,
         2,2Ø5,33,186,198,4,18Ø
KC  121Ø DATA 9,2Ø5,33,31,195,38
         ,199,6,1Ø4,1,-2,38
CI  122Ø DATA 199,6,1Ø6,1,-2,235
         ,178,38,198,6,11Ø,1
KE  123Ø DATA 1,235,17Ø,38,198,6
         ,11Ø,1,Ø,235,162,172
OO  124Ø DATA 6Ø,49,114,21,6Ø,57
         ,119,17,44,48,177,3
FM  125Ø DATA 138,224,21Ø,228,2,
         224,38,136,38,1Ø9,1,235
GC  126Ø DATA 136,186,221,4,18Ø,
         9,2Ø5,33,235,187,172,6Ø
LB  127Ø DATA 48,114,13,6Ø,51,11
         9,9,44,48,38,162,197
LO  128Ø DATA 1,233,1Ø9,255,186,
         8,5,18Ø,9,2Ø5,33,235
LM  129Ø DATA 16Ø,Ø                    ©
```

CAPUTE!

SpeedCalc Fixes

There are two errors in the DOS 3.3 listing for Apple *SpeedCalc* in the February 1986 issue (Program 1, p. 95). Lines 0FE2 and 11F2 from the listing cannot be entered as shown because smudged characters were inadvertently changed when retouched. The lines should read as follows:

```
0FE2: CA 10 EB CA 9D 00 02 E8 6E
11F2: F5 24 85 1E 20 22 0B 60 AB
```

These changes are not necessary if you have the February 1986 Apple COMPUTE! DISK; the program on disk is correct.

ProDOS users will very likely encounter the message ERROR #56 when they attempt disk operations with that version. To correct this, you need to convert the binary (BIN) format file created by "Apple MLX" into a system (SYS) format file. To do this, first use the RENAME command to give the copy of *SpeedCalc* you entered with MLX the name SPEEDCALC.MLX. (Make sure that there is no file called just SPEEDCALC on the disk.) Then enter the three commands below, each on a separate line and each followed by pressing RETURN:

```
BLOAD SPEEDCALC.MLX
CREATE SPEEDCALC,TSYS
BSAVE SPEEDCALC, A$2000,E$3D67,
  TSYS
```

The new SPEEDCALC file now on the disk should function properly. Simply enter −SPEEDCALC to start it running. *SpeedCalc* already appears as a SYS file on the COMPUTE! DISK for February, so this change is not necessary if you have the disk.

Speedy Strings For Commodore

The "Fast Disk Catalog" utility (Program 3, p. 66) from this article in the February issue does not work as listed. The SYS addresses assume that the machine language has been appended to the end of the program, as was done for Program 2. There are two possible solutions. You can change the lines below so that the machine language is POKEd in and addressed properly:

```
GM 5Ø DIMF$(MM):A=Ø
CF 1ØØ OPEN1,8,Ø,"$Ø":SYS(AA):
       CLOSE1:CLOSE15
EC 12Ø PRINTX$:PRINT"{CYN}  "R
       IGHT$(F$(C-1),2),C-8,C,
       MM-C:B=C:AA=AA+16:GOTO8Ø
XG 2ØØ POKE987,7Ø:POKE988,Ø:SY
       S(AD)
```

Alternatively, you can append the machine language to the end of the program as was done for Program 2. This results in a shorter program that runs faster. To do this, *don't change any lines in the program as listed*. Instead, add the lines shown below, then type RUN 500. When the program ends, delete line 25 and all lines above 330, then immediately save a copy of the revised program.

```
GJ 5ØØ POKE 45,(PEEK(45)+117)A
       ND 255:POKE 46,PEEK(46)
       +1-(PEEK(45)<117)
KC 51Ø POKE 47,PEEK(45):POKE 4
       8,PEEK(46):POKE 49,PEEK
       (45):POKE 5Ø,PEEK(46)
GJ 52Ø RESTORE:AD=PEEK(45)+256
       *PEEK(46)-373:FOR I=Ø T
       O 367:READ D:POKE AD+I,
       D:NEXT
GK 53Ø PRINT"{2 DOWN}DELETE LI
       NE 25 AND ALL LINES ABO
       VE 33Ø, THEN SAVE NEW V
       ERSION":END
KC 11Ø5 DATA Ø                  ©
```

The World Inside the Computer

Fred D'Ignazio, Associate Editor

The Robot Inside You

Why are children so fascinated with robots? For that matter, why is everyone so fascinated with robots? The answer is that robots seem the most lifelike of all machines, and the most like real people.

When we see a little robot "toddler" like HEROjr sing songs to a trashcan, or when we watch a Movit robot like the WAO (pronounced "Wow") skitter crablike around the kitchen floor, avoiding tables and gargantuan human feet, we feel an uncanny thrill, as if we are watching a minor miracle. We know that these little machines are not alive. But they are sending visual cues of "aliveness" to the deepest parts of our brain. And these visual processing centers are flashing the message "Alive! Alive!" to the higher-level, rational center of our brain. We can deny that the machines are alive, but we will continue to feel that somehow they really are.

Young children most strongly and visibly reflect this sense of the aliveness of robots. Children's unfettered imaginations and their incomplete mastery of the scientific view of the world (so ingrained in us adults) cause them to see all sorts of objects as being alive—including teddy bears, dolls, shadows, imaginary friends, and, of course, robots. For them, the logic is simple: If it seems alive and acts alive, then it must be alive.

Not only do children ascribe the quality of aliveness to an object based on its behavior, but they also project a psychology—a personality or character—into the object. The object's personality stems partly from its behavior (if a robot sings to a trashcan, it must be a "silly" robot), but also as a projection of children's own personalities—their wishes, dreams, fears, and subconscious feelings. It would be impossible for children to articulate what these feelings and attributes are, but they sense them immediately in a teddy bear, a beloved blanket, or an animated little robot.

The being that children see in these objects is very real, since it is a part of themselves. It may be their dark side, light side, or their happy or sad side, but it is an expression of a dimension of their own personality. Collectively these dimensions form children's complex, often contradictory humanity. In a real sense, then, there is a robot—a multitude of robots—inside every child; indeed, there are robots inside every one of us.

It is interesting to watch children struggle with the "Is it alive or not?" dilemma presented by today's robots and lifelike computer programs because we will all soon be facing this dilemma. In the coming years we adults will find our rational, scientific view of machines and other nonliving objects challenged by their increasingly lifelike characteristics. Their speech, mobility, sense of the world around them, and lifelike response are improving rapidly. All these traits will soon offer compelling evidence to our subconscious that the machines are really alive.

Kids feel this way already. For example, one little neighborhood boy of COMPUTE! staffer Debi Nash played the new Activision game "Modern Computer People" in which little beings live inside the computer and interact with the world outside. The boy believed in the little creatures, and came by the Nash's house every day to talk with them and watch them live their lives. Unfortunately, one of the little people began to overeat. No matter what Debi and her family did, he kept stuffing himself. Suddenly the program crashed, and the person disappeared. Debi told me that telling the boy about the person's demise was as hard to do as telling him that one of his friends had died.

Here in Birmingham, my six-year-old son Eric recently spent a couple days with A.G. Bear from Axlon Corp. A.G. talks in bear language but mimics human speech tones and rhythms with a little microchip in a voice box inside his chest. When Eric took A.G. to bed the first night, he had to take the voice box out of A.G. and leave it on the coffee table in the living room. Otherwise, A.G. would have begun talking every time Eric rolled over in bed or muttered something in his sleep. Eric happily took the bear to bed with him, but before he did he rushed over to the coffee table and wished the voice box good night. As I watched this little ritual from across the living room, I had the weird feeling that, for Eric, the voice box somehow held the little bear's electronic soul.

Last week, my nine-year-old daughter Catie and I were at the Bits & Bytes Computer Show for Children in Dallas, Texas. Together, she and I spoke to almost 400 schoolchildren about "Robot Pets & Friends." We demonstrated several popular robots, including Omnibot 2000, the Movit Family, and HEROjr, and we held a "Design Your Own Robot" contest which Catie judged.

The children's robot designs were original, diverse, and complex. To some extent, they resembled the robots that Catie and I had demonstrated and the robots of popular movies and TV shows. To a much greater extent, however, they were reflections of the children's own personalities. They were a revealing glimpse of the robots that dwell inside all of us. ©

Computers and Society

David D Thornburg, Associate Editor

Humanizing The User Interface, Part 2

Last month I wrote about several ways to make software easy to use. Now let's look at a model of human behavior that may hold the key for those who want to make computer programs that really stand apart from the crowd.

Psychologists and sociologists have spent a lot of time trying to figure out why people perform activities like rock climbing, playing chess, and performing other tasks for which the motivation appears to be internal (or *autotelic*). One behavioral scientist who has studied this area in some depth is Mihaly Csikszentmihalyi, whose book, "Beyond Boredom and Anxiety" (Josey-Bass, Inc., 1975), makes some observations that are of great value to those of us involved with the design of user interfaces.

I believe that a good user interface makes the computer transparent to the user so he or she is free to interact purely with the application. Now look at some comments (typical of many) that Csikszentmihalyi collected from people involved with activities like playing chess: "The game is a struggle, and the concentration is like breathing—you never think of it. The roof could fall in and, if it missed you, you would be unaware of it."

A rock climber said: "You are so involved in what you are doing that you aren't thinking of yourself as separate from the immediate activity. You don't see yourself as separate from what you are doing."

The people described above are in "flow"—a state where action and awareness have merged. In this state the connection between the participant and the activity is so close that everything else seems to disappear.

Each of us has experienced flow at one time or another—perhaps while playing a game or watching a movie. But flow is a hard state to maintain. For example, a computer user might experience flow with a video game, only to be bumped out of this state by the computer not responding fast enough, or by requiring the entry of a complex command that needs to be thought about consciously.

People who are not in a state of flow are often in a state of boredom or anxiety. Flow appears as a narrow band between the two.

When a person feels that the complexity of a task is too great for his or her skill level, a state of anxiety is produced. On the other hand, someone who has a lot of skill will be bored if the challenges are not great enough. Flow exists when the complexity of a task is appropriate for the skill of the participant.

As many readers will attest, good video games provide a nice model of the flow state. Most games allow the player to progress through a series of levels. The first level may induce anxiety in the novice player, but the player then acquires enough skill to enter a state of flow. The trick in good game design is to progressively increase the challenge level so that, as the player becomes more skilled, boredom doesn't set in.

This same model could be applied to the design of other types of programs. For example, a full-featured word processor might start out by encouraging the user to work with only a limited set of features, making the product easy to learn. As the user's skill increases more and more features can be revealed until the product is mastered.

This model of the flow state can help us understand another aspect of program design that seems to be misunderstood by some people. There is a difference between making a product easy to learn and making it easy to use. Ideally the product should have both of these features. Instead we often encounter programs that provide a tremendous amount of hand-holding for the neophyte, but which are cumbersome to use once the product is mastered. Ease of learning, in this case, makes the product cumbersome to the proficient user.

Many of the complaints that have been lodged against the early releases of Macintosh software can be traced to this conflict. The use of pull-down menus is wonderful to the first-time user, since various options and commands can be presented in plain English. However, the physical act of moving the mouse to the menu bar, opening the menu, moving the mouse to the desired selection, and selecting this item, is cumbersome to the user who already knows what choice he or she wants to make. This is why an increasing number of Macintosh programs have single keystroke equivalent commands for menu selections. For example, a menu item named SAVE is a more obvious command than Ctrl-S, but the latter is an easier command to give, once the user knows its meaning.

Ideally, programs should provide a seamless transition from being easy to learn to being easy to use. Each computer user should be free to learn at his or her own rate.

In the past, such programs were hard to create, given the limitations of the computers on which they were running. Now that the Macintosh and the Amiga have entered the scene, computer horsepower is no longer an issue. The time has come for computers to become as easy to use as any other appliance we have at home. ©

The Beginners Page

Tom R Halfhill, Editor

More String-Slicing

Last month we saw how you can copy pieces of character strings using the LEFT$ and RIGHT$ functions found in versions of Microsoft BASIC. For even more flexibility, most Microsoft-style BASICs include a third function for extracting sections of strings. Called MID$ ("mid-string"), this function lets you copy a section from the middle of a string.

The basic format is MID$ (*string$*,*n1*,*n2*), where *string$* is a string variable or literal string; *n1* is a number representing the beginning character position of the substring you want to extract; and *n2* is a number representing the number of characters in the substring you want to extract. For example:

```
10 A$="JAMES FENIMORE COOPER"
20 PRINT MID$(A$,7,8)
30 B$=MID$(A$,11,4)
40 PRINT B$
50 PRINT A$
```

When you run this program, the result is:

```
FENIMORE
MORE
JAMES FENIMORE COOPER
```

Line 20 prints the eight characters starting at position seven in A$, resulting in the substring FENIMORE. (Remember that spaces count as characters.) Lines 30 and 40 do much the same thing, but copy the four characters starting at position 11 into the string variable B$ before printing them out. This method is useful if you need to print B$ later in your program or manipulate B$ in some other way. Line 50 shows that the MID$ function, like LEFT$ and RIGHT$, does not disturb the original contents of A$.

MID$ is handy for so many different things that it's hard to come up with a generalized example. It can even be used to replace LEFT$ and RIGHT$—for instance, MID$(A$,1,10) is equivalent to LEFT$(A$,10), and MID$(A$,LEN(A$)--9,LEN(A$)) is the same as RIGHT$(A$,10). One useful application of MID$ is to store a bunch of short strings as a single long string, then pick out the substring you want with MID$. For example, let's say you're writing some sort of program that needs to print out the months of the year, perhaps as labels for a budget or chart. You could abbreviate the names of the months as equal-length substrings within one large string, like this:

```
10 M$="JANFEBMARAPRMAYJUNJUL
  AUGSEPOCTNOVDEC"
```

Now suppose that the numeric variable M contains the number of the month you need to print out—maybe as a result of an INPUT statement:

```
20 PRINT "WHICH MONTH TO
  PRINT";
30 INPUT M
40 PRINT MID$(M$,M*3−2,3)
```

Depending on the user's response (1 for January, 2 for February, etc.), line 40 prints out the proper month name. Or you could print out all the months with a loop—FOR M=1 TO 12:PRINT MID$(M$,M*3−2,3) :NEXT M.

Storing all the months in a single string and extracting the one you want with MID$ is more efficient than using 12 separate strings for the same purpose. It's also more efficient in some ways than a string array (a subject we'll cover in a future column).

Atari And TI Strings

There are no LEFT$, RIGHT$, or MID$ functions in TI BASIC or the Atari BASIC found on Atari 400/800, XL, and XE computers. These BASICs handle strings a little differently than Microsoft BASIC does. (Note that Microsoft BASIC is available on cartridge for Atari computers, and some BASICs available from independent suppliers also support Microsoft-style strings.)

TI BASIC's statement for segmenting strings is SEG$. It works exactly like MID$ in Microsoft BASIC—the statement B$=SEG$ (A$,11,4) is equivalent to B$=MID$(A$,11,4). You can simulate LEFT$ with a statement in the form SEG$(*string$*,*n1*,*n2*), where *string$* is the string you wish to manipulate, *n1* is the starting character position of the segment within the string, and *n2* is the number of characters you wish to print or copy. For example, the statement B$=LEFT$(A$,6) can be replaced with B$=SEG$(A$,1,6).

Simulating RIGHT$ is a bit more complicated. You need a statement in the form SEG$(*string$*,LEN(*string$*)−*n1*,*n2*), where *n2* is the number of characters you wish to print or copy, and *n1* is *n2*−1. For example, B$=RIGHT$(A$,6) can be replaced with B$=SEG$ (A$,LEN(A$)−5,6).

Atari BASIC requires the same sort of manipulations. To print or copy any substring in Atari BASIC, simply specify the starting and ending character positions of the substring within the larger string. To translate B$=LEFT$(A$,6), use B$=A$(1,6). To simulate RIGHT$, use a statement in the form *string$*(LEN(*string$*)−*n*,LEN(*string$*)), where *string$* is the string you're manipulating and *n* is the number of characters you wish to print or copy minus one. For instance, to translate B$=RIGHT$(A$,6), use B$=A$(LEN(A$)−5,LEN(A$)). To simulate MID$, use the statement *string$*(*n1*,*n2*), where *n1* is the starting character position (just like MID$), and *n2* equals *n1* plus the number of characters you wish to print or copy minus one. Thus, the Microsoft statement B$=MID$ (A$,11,4) is translated as B$=A$ (11,14). ©

Telecomputing Today

Arlan R. Levitan

An April Trade Show Report

While most of the computer press converged on Lost Wages, Nevada in November of 1985 to attend yet another humdrum computer trade show, this columnist packed up his gear and headed for a counterculture communications fest held on the multileveled U.S.S. Flotsam, an ex-petro supertanker converted for use as a floating convention facility. Dubbed COMMDECKS 85 by show sponsor Aski Blok, it provided a fresh look at the lunatic fringe of computer communications.

There had been some doubt as to whether the show would get off the ground at all. Picket lines were set up by angry labor protesters who had been written bad redundancy checks by the show sponsor. The strikers were demanding even parity for all data transmitted to and from the show, making it almost impossible for exhibitors to set up their tables in the days preceding the show's opening. In the end, the demonstrators dropped their parity demands in exchange for 14 percent more than they had been receiving, plus additional time off in the form of one extra data bit and two stop bits.

It's hard to pinpoint the most memorable products of the show (since I spent most of my time recovering from the hors d'oeuvres and beverages served at evening press conferences), but I owe it to the readers of this column (and to the IRS) to take a shot at it.

Don't Just Ask For A Light

For the health-minded telecomputerist, Natural Language's line of optical wave modems are the first of the new "light" modems, transmitting 30 percent fewer characters than their wire-based counterparts. The new units are also said to aid the digestion of serial data (a.k.a. "number crunching") due to their high fiber optic content.

Setting a hard standard to beat for intelligent modems is Thought System's new Kreskin 2400. How smart is it? The Kreskin reportedly can detect a busy signal before a call is actually made. Some recent prototypes also refuse to dial a remote Bulletin Board System if the unit's advanced circuitry senses there is nothing interesting to read or download on the BBS. The heart of the Kreskin is a superfast proprietary CPU chip capable of executing an infinite loop in 37 seconds. It translates the incoming stream of data into your choice of French, Italian, or Chinese (English is an extra-cost option).

No trade show would be complete without the obligatory raft of seminars and workshops, and COMMDECKS was no exception. Things did get off to a confusing start, however, when the kickoff session "The Future of VideoTex" turned out to be a panel discussion on merchandising VCRs and TVs in Dallas, Fort Worth, Austin, and Houston.

The highlight of the show's conference schedule had to be "Null Modems—Threat or Menace?"—a discussion of the trend toward violence in data communications hardware. The introduction of the Ninja, Terminator, and Rambo class of modem eliminators at the show further fueled the controversy to new heights.

Micro Telecomputing

With the price of mobile telephones dropping faster than a brick, Phylum Systems of Paramecium, California figures to cash in big with a $14.95 limited-distance mobile modem dubbed the Amoeba. Phylum's vice president of marketing, Ernest Flagella, says the single cellular unit will be shipping either "(1) real soon now, (2) in two weeks, (3) when the manual comes back from the printer, or (4) when Atari ST and Amiga owners stop bickering over who bought the better machine."

Meanwhile, the Arapaho Indian Nation is entering the packet-switching network race to serve the communications needs of telecomputerists in the remote West. Bowing to the pressure of environmentalist groups, the Arapaho elders have agreed to house their telecommunications equipment inside structures disguised as totem poles. The job of cabling the poles has been awarded to RS-232 ace Louie "Bent Pin" Carson. Although Carson anticipates a high degree of difficulty in routing the cables within the highly confined spaces of the totems, he feels that a shot at everlasting fame is worth all of the headaches. When the job is done, Carson will have become the first man to wire a head for a reservation.

After-hours entertainment got physical on Friday night as anybaudy who is anybaudy attended a sports competition for manufacturers of multiuser LANs dubbed "Battle of the Network Stars." Over 50 teams vied for the coveted "Lord of the Rings" title, and the highly favored New York Subcarriers were disqualified in the early rounds for passing bad tokens in the relay.

To be perfectly honest, attendance at COMMDECKS 85 was far below the anticipated crowd of 25,000+. While hanging over the quarterdeck railing on the third day of the show, I bumped into promoter Aski Blok once more and quizzed him about the low number of attendees. "Well, it's really not too bad if you take everything into account," he said. "Our current location is kind of hard for people to get to. I think it would have been a lot more crowded if we hadn't cast off from the docks for the open C—the C programming workshop, that is." ©

Creating Rhythms

A year ago, in the March 1985 issue of COMPUTE!, I published a program called "Drum Practice" for the TI-99/4A. That program was limited to quarter notes and quarter rests and the rhythms listed in DATA statements. This month, I'm offering a more complex program.You can create the rhythm for one measure by choosing notes and rests, and then the computer will play the rhythm for eight measures.

Lines 110–190 print the instructions. The different kinds of notes and rests available will appear at the bottom of the screen. You can use the arrow keys (on S and D) to move the red marker left or right to make your selection, then press the ENTER key. Your choice will then be printed on the staff above.

The available notes are a quarter note, two eighth notes together, one eighth note, two sixteenth notes, a dotted eighth note with a sixteenth note, a quarter rest, and an eighth rest.

Line 200 sets the time T equal to 75. If you want the rhythm to play faster decrease this number which represents the duration of a sixteenth note. Lines 210–240 read in from data (lines 260–330) the definitions for the graphic characters and define the characters from numbers 91 to 128. Line 340 defines R for a row number for the staff. Lines 350–360 define the red arrow used as a marker under the notes to be selected.

Lines 370–390 define variables in an array for the seven possible choices. Line 400 contains the data for this loop. For each of the choices from 1 to 7, D(C) is a value representing the counts—4 for one count, 2 for a half count. This variable is used to make sure the user makes a valid choice. For example, the computer will not allow a quarter note to be chosen if only a half of a count is left in the measure. COL(C) is the column and is used to place the red marker.

The Rhythm Track

S$(C) represents the durations when the rhythm is played. A sixteenth note factor is 1, so the quarter note is 4. Two eighth notes are 22, and one eighth note is 2. The two sixteenth notes are 11, and a dotted eighth with a sixteenth are 31. The rests are W and H. As the notes and rests are chosen, the string RHY$ will add on values of S$ (line 870).

Lines 1250–1390 play the rhythm. Line 1270 finds the length L of the string RHY$. Line 1280 starts the loop for L number of times. Line 1290 looks at one character at a time of RHY$. If the character A$ is a letter, a rest is indicated so a frequency of 9999 with a volume of 30 is used. If A$ is a number, that number is used as a factor times the previously defined T for the duration in the CALL SOUND statement, line 1310. If you prefer a different sound, change the frequency numbers in line 1310. I used the noise of −5 plus the frequency of 330. Line 1370 stops the sound so you can hear the different notes. The measure is played eight times.

Lines 420–440 wait for you to press ENTER before the program continues. Lines 460–500 clear the screen and print the notes using the redefined symbols. The lowercase y and z are typed by releasing the ALPHA LOCK key. Most of the symbols are typed by using the function key. Lines 510–630 draw the staff.

Lines 650–690 initialize variables for choosing the notes. COUNT and CHECK are used to determine how many notes and rests can be used in the measure. This measure is 4/4 time. A sixteenth note has a value of 1, so the COUNT will go up to 16. CHECK is how many points are remaining in the measure. These numbers are used to verify which notes and rests can be used in the measure.

PLACE is the column number where the note or rest will start being drawn on the staff. The first note will start in column 8. PLACE is incremented depending on which note or set of notes is chosen. Lines 690–820 are the lines to get the user's choice. Line 730 makes sure the left arrow key (S), the right arrow key (D), or the ENTER key is chosen; all other keys are ignored by branching back to the CALL KEY statement. C is the choice number, and COL(C) is the column where the red marker appears for the choice.

It's Timing That Counts

Line 830 makes sure the choice is valid. The D timing value must be less than or equal to the number of sixteenth counts available. If the choice is not valid, the program plays an "uh-oh" sound and branches back to line 720, which is the CALL KEY statement to get another choice. Line 870 increments the RHY$ string with the appropriate timing factors. Line 880 branches to the proper place for drawing the notes or rest and incrementing PLACE.

Lines 1220–1240 increment the COUNT and recalculate the CHECK time. If the measure is not full, the program goes back to get another choice. Lines 1250–1390 play the measure eight times. To stop the program, press FCTN BREAK.

If you have trouble running this program and get an error message in 220, 230, or 380, the actual cause of the error is most likely in the DATA statements of lines 260–330 or line 400.

All notes are placed at the E space of the staff, representing a snare drum rhythm. You may add to the program by including bass drum notes, cymbal rhythms, and tom-toms. To use this program for a melody instrument, you can use the up and down keys to move the note on the staff, then use a variable frequency to play the note.

You may use the general idea of this program in choosing items to go with a different theme of graphics, not music—perhaps building a game or drawing a picture by choosing different shapes.

Rhythms

```
100 REM  RHYTHMS
11Ø CALL CLEAR
12Ø PRINT TAB(1Ø);"RHYTHMS"
13Ø PRINT ::"CHOOSE THE NOT
    ES FROM THE"
14Ø PRINT :"BOTTOM OF THE S
    CREEN BY"
15Ø PRINT :"USING THE LEFT
    AND RIGHT"
16Ø PRINT :"ARROW KEYS TO M
    OVE AND"
17Ø PRINT :"THE ENTER KEY T
    O SELECT."
18Ø PRINT ::"WHEN THE MEASU
    RE IS COMPLETE"
19Ø PRINT :"YOU WILL HEAR T
    HE RHYTHM."
2ØØ T=75
21Ø FOR C=91 TO 128
22Ø READ C$
23Ø CALL CHAR(C,C$)
24Ø NEXT C
25Ø REM  DATA FOR CHARACTER
    S
26Ø DATA ØØØØØFØ8ØFØ8Ø8Ø8,Ø
    ØØØF8Ø8F8Ø8Ø8Ø8,Ø8Ø8Ø83
    8F9E8,ØØØØFFØ1Ø7Ø1Ø1Ø1,
    1ØØ8ØC183Ø3ØØ8Ø4,1C2Ø2Ø
    1ØØ8
27Ø DATA ØØØØØØØØØØØØFFØØ,0
    ØØØFF0ØØ4DØFFØØ,D4ØØFFD
    ØØØØØFF,ØØØØØØØØØØØØFF,
    ØØØØFFØØØØØØFF,ØØØØØØØØ
    ØØØØFFØ8
28Ø DATA Ø8Ø8FFØ828Ø8FFØ8,2
    8Ø8FFØ8Ø8Ø8FF,ØØØØØØØØ
    ØØØFF1Ø,2424447EØ4Ø4FF1
    ,1Ø2424447EØ4FF,ØØØØØ8Ø
    8Ø8Ø8FFØ8
29Ø REM
3ØØ DATA Ø8Ø8FF78F87ØFF,ØØØ
    ØØFØ8Ø8Ø8FFØ8,ØØØØF8Ø8Ø
    8Ø8FFØ8,ØØØØØ8ØCØAØ9FFØ
    8,ØØØØØFØ8ØFØ8FFØ8
31Ø DATA ØØØØF8Ø8F8Ø8FFØ8,Ø
    ØØØFFØ1Ø7Ø1FFØ1,ØØØØFF6
    27EØ4FF1,1Ø2ØFFØØØØØØFF
    ,Ø8Ø8FF78F978FF,1ØØ8ØC1
    83Ø3ØØ8Ø4
32Ø DATA 1C2Ø2Ø1ØØ8,ØØØØØ8Ø
    8Ø8Ø8Ø8Ø8,Ø8Ø8Ø878F87,Ø
    ØØØØFØ8Ø8Ø8Ø8Ø8,ØØØØF8Ø
    8Ø8Ø8Ø8Ø8,ØØØØØ8ØCØAØ9Ø
    8Ø8
33Ø DATA 627EØ2Ø4Ø81Ø2,Ø1Ø1
    FFØF1FØEFF,Ø1Ø1Ø1ØF1FØE
34Ø R=5
35Ø CALL CHAR(136,"1Ø387CFE
    1Ø1Ø1Ø1")
36Ø CALL COLOR(14,7,1)
37Ø FOR C=1 TO 7
38Ø READ D(C),COL(C),S$(C)
39Ø NEXT C
4ØØ DATA 4,4,4,4,8,22,2,12,
    2,2,15,11,4,19,31,4,24,
    W,2,28,H
41Ø REM
42Ø PRINT ::"PRESS <ENTER>
    TO START."
43Ø CALL KEY(Ø,K,S)
44Ø IF K<>13 THEN 43Ø
45Ø REM  DRAW NOTES
46Ø CALL CLEAR
47Ø PRINT " y   {|   }  [\
    {^   _"
48Ø PRINT " z   zz  z  zz
    zz   '   ~"
49Ø CALL HCHAR(23.2Ø,128)
5ØØ PRINT
51Ø REM  DRAW STAFF
52Ø CALL HCHAR(R,3,97)
53Ø CALL HCHAR(R  ,3,98)
54Ø CALL HCHAR(R  ,3,99)
55Ø CALL HCHAR(R,4,1ØØ,22)
56Ø CALL HCHAR(R+1,4,1Ø1,22
    )
57Ø CALL HCHAR(R+2,4,1Ø1,22
    )
58Ø CALL HCHAR(R,26,1Ø2)
59Ø CALL HCHAR(R+1,26,1Ø3)
6ØØ CALL HCHAR(R+2,26,1Ø4)
61Ø CALL HCHAR(R,5,1Ø5)
62Ø CALL HCHAR(R+1,5,1Ø6)
63Ø CALL HCHAR(R+2,5,1Ø7)
64Ø REM  CHOOSE NOTES
65Ø COUNT=Ø
66Ø CHECK=16
67Ø PLACE=8
68Ø RHY$=""
69Ø C=1
7ØØ CALL HCHAR(23,COL(C),13
    6)
71Ø CALL SOUND(1ØØ,14ØØ,2)
72Ø CALL KEY(Ø,K,S)
73Ø IF (K<>13)*(K<>83)*(K<>
    68)THEN 72Ø
74Ø CALL HCHAR(23,COL(C),32
    )
75Ø IF K<>83 THEN 78Ø
76Ø C=C-1
77Ø IF C>=1 THEN 7ØØ ELSE 6
    9Ø
78Ø IF K<>68 THEN 83Ø
79Ø C=C+1
8ØØ IF C<=7 THEN 7ØØ
81Ø C=7
82Ø GOTO 7ØØ
83Ø IF D(C)<=CHECK THEN 87Ø
84Ø CALL SOUND(15Ø,33Ø,2)
85Ø CALL SOUND(15Ø,262,2)
86Ø GOTO 7ØØ
87Ø RHY$=RHY$&S$(C)
88Ø ON C GOTO 89Ø,93Ø,99Ø,1
    Ø3Ø,1Ø9Ø,115Ø,119Ø
89Ø CALL HCHAR(R,PLACE,1Ø8)
9ØØ CALL HCHAR(R+1,PLACE,1Ø
    9)
91Ø PLACE=PLACE+3
92Ø GOTO 122Ø
93Ø CALL HCHAR(R,PLACE,11Ø)
94Ø CALL HCHAR(R+1,PLACE,1Ø
    9)
95Ø CALL HCHAR(R,PLACE+1,11
    1)
96Ø CALL HCHAR(R+1,PLACE+1,
    1Ø9)
97Ø PLACE=PLACE+3
98Ø GOTO 122Ø
99Ø CALL HCHAR(R,PLACE,112)
1ØØØ CALL HCHAR(R+1,PLACE,1
     Ø9)
1Ø1Ø PLACE=PLACE+2
1Ø2Ø GOTO 122Ø
1Ø3Ø CALL HCHAR(R,PLACE,113
     )
1Ø4Ø CALL HCHAR(R+1,PLACE,1
     Ø9)
1Ø5Ø CALL HCHAR(R,PLACE+1,1
     14)
1Ø6Ø CALL HCHAR(R+1,PLACE+1
     ,1Ø9)
1Ø7Ø PLACE=PLACE+2
1Ø8Ø GOTO 122Ø
1Ø9Ø CALL HCHAR(R,PLACE,11Ø
     )
11ØØ CALL HCHAR(R+1,PLACE,1
     18)
111Ø CALL HCHAR(R,PLACE+1,1
     15)
112Ø CALL HCHAR(R+1,PLACE+1
     ,127)
113Ø PLACE=PLACE+3
114Ø GOTO 122Ø
115Ø CALL HCHAR(R+1,PLACE,1
     19)
116Ø CALL HCHAR(R+2,PLACE,1
     2Ø)
117Ø PLACE=PLACE+3
118Ø GOTO 122Ø
119Ø CALL HCHAR(R+1,PLACE,1
     16)
12ØØ CALL HCHAR(R+2,PLACE,1
     17)
121Ø PLACE=PLACE+2
122Ø COUNT=COUNT+D(C)
123Ø CHECK=16-COUNT
124Ø IF COUNT<16 THEN 7ØØ
125Ø REM  PLAY RHYTHM
126Ø FOR TIME=1 TO 8
127Ø L=LEN(RHY$)
128Ø FOR M=1 TO L
129Ø A$=SEG$(RHY$,M,1)
13ØØ IF (A$="W")+(A$="H")TH
     EN 133Ø
131Ø CALL SOUND(T*VAL(A$),-
     5,2,33Ø,4)
132Ø GOTO 137Ø
133Ø REST=T
134Ø IF A$="H" THEN 136Ø
135Ø REST=REST*2
136Ø CALL SOUND(REST,9999,3
     Ø)
137Ø CALL SOUND(1,9999,3Ø)
138Ø NEXT M
139Ø NEXT TIME
14ØØ FOR DEL=1 TO 5ØØ
141Ø NEXT DEL
142Ø GOTO 46Ø
143Ø END                  ©
```

Attention Programmers

COMPUTE! magazine is currently looking for quality articles on Commodore, Atari, Apple, and IBM computers (including the Commodore Amiga and Atari ST). If you have an interesting home application, educational program, programming utility, or game, submit it to COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. Or write for a copy of our "Writer's Guidelines."

IBM Personal Computing

Donald B. Trivette

Two Checkers And A Manager

Anyone who spells as badly as I do is bound to love spelling checkers—and here are two new products that are getting a lot of attention.

Borland, the folks who brought you *Turbo Pascal* and the popular *SideKick*, have come up with another product headed for the best-seller's shelf. *Turbo Lightning* is a memory-resident spelling checker—it monitors every word you type and instantly beeps when you've made a mistake. Then, by pressing a key, you can call forth (in a box superimposed over your text) a list of the most likely correct words. It uses the 83,000-word Random House dictionary as its spelling authority. *Lightning* also has a thesaurus option which lets you select just the right word from a 50,000-word Random House thesaurus. All of this from within any program—word processor, spreadsheet, data management, or communications—just by pressing a few keys.

Here's how it works. *Lightning* stores a small dictionary in RAM. When installing the program, you must select one of three sizes: 6,000 words, 12,000 words, or 16,000 words. The larger the dictionary, the larger *Lightning*'s vocabulary, and the less often it beeps for a word that is really correct. The trade-off, as always, is memory. As you type a word, *Lightning* consults the in-memory dictionary and beeps if there is no match. At this point, you may press the Alt-F9 keys to make the program consult the larger disk-based dictionary. *Lightning* then either confirms your spelling as correct or lists possible choices based on sound-alike words.

Two different disk-based dictionaries are available: one for hard disk systems and a smaller one for floppy disk computers. Since most of us have a small working vocabulary, the scheme of a RAM dictionary supplemented by one on disk is quite workable.

A Flexible Engine

If you're thinking that a poor speller would be beeped to distraction, you are right. Fortunately, *Lightning* allows the auto-proof mode to be turned off; checking may then be requested on individual words or a screen at a time.

Borland plans to issue additional dictionaries and databases for use with the *Lightning* engine. In fact, any text-type data—even your own files—could be indexed and made accessible. *Turbo Lightning* is a sophisticated program with more potential than just a spelling checker. (*Turbo Lightning*, $99.95, Borland International, 4585 Scotts Valley Drive, Scotts Valley, CA 95066.)

The second new spelling-checker is *Reference Set* from Reference Software. It too uses the Random House dictionary and thesaurus (what happened to Webster's?). *Reference Set* doesn't check each word as you type, but rather waits for you to request a spelling check by pressing Alt-D for dictionary or Alt-T for thesaurus. A window pops up over your text showing possible correct spellings (or alternate words); pressing a key deletes the old word and inserts the new one.

Although the dictionary is referenced from disk, the program maintains an index in memory so the time to locate a word, even with floppy disk, is typically less than a second. *Reference Set* includes two different sized dictionaries and thesauri for floppy and hard disk systems. The modest *Reference Set* engine—the memory-resident program that accomplishes the lookup—uses about 20K; by comparison, *Lightning* uses about 83K. (*Reference Set* Version 1, $89.95, Reference Software, 2363 Boulevard Circle, Walnut Creek, CA 94595.) Both programs work best with a hard disk, but either may be used with a two-drive floppy system.

Automatic Stock Quotes

The "Manager" referred to in the title of this column alludes to a new program that works with the popular *Andrew Tobias' Managing Your Money* (see "IBM Personal Computing," December 1985). Called *Managing the Market*, it's a communications program that dials the Dow Jones News/Retrieval service, collects quotations, and updates the prices for the securities in an *MYM* portfolio. Pushing three or four keys dials the number, enters the password, selects the service, requests the quotes, updates the files, and disconnects. One nifty feature allows quotes to be ordered either by the percent change or by the absolute change—a real timesaver for those who monitor a lot of stocks. Output can be printed or saved for later analysis in a file readable by *Lotus 1-2-3*.

If you think this program would be too expensive to use with a modest portfolio, you may be pleasantly surprised. I've been updating about a dozen stocks, five days a week, after 6 p.m. when the rates are lowest, and the bill from Dow Jones is less than $10 a month. *Managing the Market* comes with a temporary password and one hour of free time with Dow Jones, so you can begin using it right away. Of course, you must have a modem; the program supports all Hayes and Hayes-compatible modems as well as a dozen or so other makes. (*Managing the Market*, $79.95, MECA, 12 Saugatuck Ave., Westport, CN, 06880.) ©

Bill Wilkinson

Binary Files, Unite!

I've had several people write me that various programs designed for use with binary (machine language) files don't work with Atari's *Macro Assembler* (*AMAC*), OSS's *MAC/65*, or a couple of other assemblers. Or possibly a program will work with a small binary file produced by these assemblers, but not with a larger one. Why all these problems when the simple Atari *Assembler/Editor* cartridge works so well?

The root of the problem is the Atari Disk Operating System definition of a binary file, so let's examine that first. (Besides, maybe we'll learn a few extra goodies on the way.) A legal Atari binary file has the following format:

1. A header of two bytes, each with a value of 255 (hex $FF).

2. Two more bytes indicating the starting address of a *segment* of the binary file. The two bytes are in standard 6502 low-byte/high-byte order.

3. Two more bytes indicating the ending address of that same file segment.

4. A sequence of bytes which constitute the actual binary code to be loaded into memory for the segment defined by the preceding four bytes. The number of bytes may be determined by subtracting the starting address from the ending address and then adding one.

5. If there are no further segments, there should be no more bytes in the file.

6. If there are more segments, then repeat this sequence of steps starting at either step 1 *or* step 2.

And that's it. A really neat, clean, format. Watch out for that last step, though. First, it says that the number of segments is theoretically unlimited. Second, it says that header bytes (dual hex $FF bytes) *may* occur at the start of any segment. It also implies that there is no particular order necessary to a binary file; it's perfectly OK to load the segment(s) at higher memory addresses before the one(s) at lower addresses.

RUN And INIT Vectors

Before moving on, there are two other niceties about DOS binary files worth knowing. When DOS loads a binary file (including an AUTORUN.SYS file at powerup), it monitors two locations. The simpler of the two is the RUN vector. Before DOS begins loading the binary file, it puts a known value into the two bytes at locations 736–737 (hex $2E0–$2E1). When the file is completely loaded (i.e., when DOS encounters the end of the file, step 5 above), if the contents of location 736 have been changed, then DOS assumes the new contents specify the address of the beginning of the program just loaded. DOS then calls the program (via a JSR) at that address.

The second monitored location is the INIT vector at address 738 (hex $2E2). This vector works much the same as the RUN vector, but DOS initializes and checks it for *each segment* as the segments are loaded. If the INIT vector's contents are altered, then DOS assumes the user program wants to stop the loading process for a moment, long enough to call a subroutine. So DOS calls (via a JSR) at the requested address, expecting that the subroutine will return so the loading process can continue. This is a very handy feature. Most of you have probably seen it at work, such as when you run (or boot) a program which puts up an introductory screen (maybe just a title and a PLEASE WAIT message) and then continues to load.

The other important difference between the RUN and INIT vectors is that DOS leaves channel number one open while the INIT routine is called. (DOS always opens and loads the binary file via this channel.) I suppose a really tricky program could close channel one, open up a different binary file, and then return to DOS. DOS would proceed to load the new file as if it were continuing the load of the original one. Most of the time, though, INIT routines should not touch channel one.

More On Segmented Files

Back to the main subject: Why do some programs have problems with binary files produced by some assemblers? Well, if all programs followed the complete binary file format as given by steps 1 through 6 above, there would probably be no incompatibilities. Unfortunately, many people who have used no assembler except the old cartridge have ignored segmented files. They have assumed that a binary file consists of steps 1 through 4, one time only, with a single large segment. Perhaps this is because many programmers first worked with Apple DOS, CP/M, and other operating systems with not-so-intelligent binary file formats. Or perhaps it is because the supposedly simple assembler cartridge is, in some ways, smarter than more advanced assemblers. In particular, the assembler cartridge will *not* produce multiple segments unless the programmer specifically asks for them (via an *= directive to force a change to the location counter).

Yet other assemblers (including *AMAC* and *MAC/65*) never produce a segment longer than a particular size (usually a page—256 bytes—or less). If the programmer coded a longer segment, these assemblers automatically break it up into smaller pieces. Why? Probably to gain speed and lessen the work of assembly, since the assembler cartridge is doing a lot of work remembering the ending addresses of segments.

Now, if my only concern were those few programs which don't properly load all binary files, I would simply have showed their authors the way to fix them. But there is a secondary advantage to programs which consist of larger segments: They load faster! Sometimes *much* faster. So this month I give you the BASIC program below, which takes any binary file and attempts to "unify" it. In particular, if the start address of one segment directly follows the end address of the preceding segment, they are consolidated into a single segment. And so on, so far as the space in BUF$ allows.

And, last but not least, there's another minor bonus. Often, someone who writes an assembly language program purposely leaves space to be filled in later (e.g., by a filename, counter, etc.). If this reserved space occurs in the midst of code (probably not good practice, but it happens), it forces even the assembler cartridge to break the file into segments. But if the reserved space is significantly less than a sector (say under 50 bytes or so), it may be faster to let DOS load filler bytes. So you can change the value of the variable FILL in line 1160 (to 40, perhaps), and this program will automatically generate up to the specified number of fill bytes in an effort to better unify the file.

Whew! Was this month's topic too heavy for you? Then write me (P.O. Box 710352, San Jose, CA 95071-0352) with *your* suggestions for a topic. No treatises please. One or two pages works best. Thanks.

Binary File Unifier

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

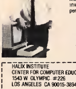
```
GG 1110 REM allocate buffer
KI 1120 REM
DI 1130 BUFSIZE=FRE(0)-300
AK 1140 OIM 8UF$(BUFS1ZE)
II 1150 OIM F1LEOLD$(40),FIL
        ENEW$(40)
KH 1200 REM
CJ 1210 REM get file name
KJ 1220 REM
NO 1230 PR1NT "I need two fi
        le names: An existin
        g"
EA 1240 PRINT "  object file
         and a new file whic
        h"
EE 1250 PR1NT "  will get th
        e 'unified' object c
        ode."
FG 1260 PR1NT
AA 1270 PRINT "Existrng file
        ? ";
DE 12B0 INPUT #16,FILEOLO$
DB 1290 PRINT "{5 SPACES}New
         file? ";
DI 1300 INPUT #16,FILENEW$
KJ 1400 REM
JC 1410 REM open files, vali
        date existing one
KL 1420 REM
FJ 1430 OPEN #1,4,0,FILEOLO$
JD 1440 GET #1,SEGLOW:GET #1
        ,SEGH1GH
KD 1450 IF SEGLOW=255 AND SE
        GHIGH=2S5 THEN 1500
PI 1460 PRINT :PR1NT "Existi
        ng file: invalid for
        mat"
KD 1470 ENO
DF 1480 REM input file okay
LC 1490 REM
GH 1500 OPEN #2,B,0,FILENEW$
MF 1510 PUT #2,SEGLOW:PUT #2
        ,SEGHIGH
KL 1600 REM
NO 1610 REM process a new or
        igin
KN 1620 REM
AK 1630 BUFPTR=0
OO 1640 BUF$=CHR$(0):8UF$(8U
        FSIZE)=CHR$(0)
HB 1650 BUF$(2)=BUF$:REM zap
         buffer
ML 1660 PUT #2,SEGLOW:PUT #2
        ,SEGH1GH
KM 1700 REM
AA 1710 REM process a segmen
        t
KD 1720 REM
IF 1730 GET #1,ENDLOW:GET #1
        ,ENOHIGH
BH 1740 SEGSTART=SEGLOW+2S6*
        SEGHIGH:SEGENO=ENDLO
        W+256*ENOHIGH
HE 1750 SEGLEN=SEGENO-SEGSTA
        RT+1
HF 1760 REM read segment int
        o buffer
ML 1770 FOR PTR=1 TO SEGLEN
KH 1780 GET #1,8YTE:8UF$(BUF
        PTR+PTR)=CHR$(BYTE)
AG 1790 NEXT PTR
KN 1B00 REM
MF 1810 REM check head of ne
        xt segment
KP 1820 REM
JG 1830 GET #1,SEGLOW:GET #1
        ,SEGH1GH
KK 1840 IF SEGLOW=2S5 AND SE
        GHIGH=255 THEN GET #
        1,SEGLOW:GET #1,SEGH
        IGH
OL 1B50 SEGNEXT=SEGLOW+256*S
        EGH1GH
ED 1860 GAP=SEGNEXT-SEGEND-1
HC 1B70 IF GAP>F1LL OR GAP<0
         THEN 2000
KA 1BB0 BUFPTR=BUFPTR+SEGLEN
        +GAP
ED 1B90 1F BUFPTR+256>8UFS1Z
        E THEN 2000
ML 1900 GOTO 1700
KG 2000 REM
DJ 2010 REM need to dump buf
        fer to
LA 2020 REM prepare for new
        origin
KJ 2030 REM
LE 2040 PUT #2,ENDLOW:PUT #2
        ,ENDHIGH
OG 2050 FOR PTR=1 TO LEN(BUF
        $)
LC 2060 PUT #2,ASC(BUF$(PTR)
        )
PO 2070 NEXT PTR
```

©

Write computer programs at home...

BE SUCCESSFUL WITH COMPUTERS

Self-paced, one-on-one instruction



Unique study program includes word-processing bookkeeping, data base management mailing lists, inventories and all about personal computers

No experience needed! Learn even before you decide on a computer.

Professional home and business applications

HALIX INSTITUTE
CENTER FOR COMPUTER EDUCATION DEPT 61 4
1543 W OLYMPIC #226
LOS ANGELES CA 90015-3894

YES! Send me free information on how I can learn about computers and programming at home!

Name ____________

Address ____________

City ____________ State/Zip ____________

PREMIUM QUALITY! LIFETIME WARRANTY!

DISK SALE!

from 33¢

Made by top USA makers, not low-end or seconds. We buy truckloads of major makers' overproduction. We can't print the maker, but you'll recognize them as among the HIGHEST QUALITY PREMIUM DISKS MADE! Certified, guaranteed 100% error free. MONEY BACK SATISFACTION GUARANTEE!

Prices are per disk	BULK					BOXED				
	100	500	1K	5K	10K	10	100	500	1K	5K
SS/DD 35trk for Apple II series #UDA	45¢	42¢	39¢	36¢	33¢	51¢	53¢	50¢	47¢	44¢
SS/DD for Single side computers #UD1	59¢	57¢	55¢	50¢	48¢	75¢	67¢	65¢	63¢	57¢
DS/DD for IBM-PC & compatibles #UD2	67¢	60¢	58¢	56¢	54¢	69¢	75¢	68¢	66¢	64¢
DS/QUAD DENSITY (single) #UD4	1.09	1.04	99¢	92¢	89¢	1.15	1.11	1.06	1.02	99¢
DS/HD "HI-DENSITY" for IBM-AT #UAT	2.15	2.12	2.10	2.05	1.99	2.40	2.25	2.22	2.20	2.18
3½" single side #UD3	1.99	1.93	1.90	1.87	1.84	1.99	1.96	1.93	1.90	1.87
3½" double side #UD6	2.85	2.55	2.45	2.40	2.38	2.80	2.70	2.55	2.48	2.40
8" SS/SD soft sect. #UD8						1.20	1.00	1.50	1.90	1.00
5¼" SLEEVES #24 white wove #SLV	4¢	3.5¢	3¢	2.8¢	2.6¢	Specify BULK or BOXED when ordering disks				
5¼" SLEEVES white tyvek #TYV	6¢	5¢	4.5¢	4.3¢	4.1¢					

INCREDIBLE PRICES ON COMPUTER ACCESSORIES!

UNIFILE-100 holds 100 disks, with lock & key, removeable top. $13.95
UNIFILE-70 Holds 70 disks, like "Flip-n-file." #FILE70 $8.95
UNIFILE-10 LIBRARY CASES, holds 10 disks, clear plastic. #FILE10 99¢
UNIPAK 5¼" DISK MAILERS, rigid cardboard, up to 3 disks #UNIPAK 10 for $5
LETTER QUAL COMPUTER PAPER, 2500 sheets,20# microperf 8½x11" #LQP $29
DRAFT QUAL COMPUTER PAPER, 3300 sheets,15# std-perf 8½x11" #DQP $29

HOW TO ORDER: Pay by MC-Visa-Amex-COD, or send check with order, minimum order $20, add $3 for shipping (call for shipping rates on paper). We can ship open account to schools with good credit, minimum purchase order $100, FOB Unitech. All orders must include daytime phone and STREET address. Money-back 30 day satisfaction guarantee. Send for FREE CATALOG listing 100's of discount computer supplies!

TOLL FREE **(800)343-0472**

UNITECH

IN MASS. (617)"UNI-TECH"
20u HUBLEY ST.
CAMBRIDGE, MA 02141

ATTENTION T.I. 99/4A OWNERS

- Diskettes - 59¢ each! Your choice SS or DD
- 512K Now Available for the 99/4A!
- 99/8 Level 4 Computer Upgrade Now Available
- Over 1500 Hardware and Software Accessories at Similar Savings

THE WORLD'S LARGEST COMPUTER ASSISTANCE GROUP

Now serving over 35,000 members worldwide with the best in technical assistance, service, and products for the Texas Instrument 99/4A Computer

To become a member and receive newsletters, catalog, technical assistance and membership package, send $10.00 for a ONE Year Membership to:

99/4A National Assistance Group
National Headquarters
P.O. Box 290812
Ft. Lauderdale, Florida 33329
Attention Membership Division
For Further Information Call 24 Hours
(305) 583-0467

Interactive Science Fiction Game For Commodore, Apple

PSI-5 Trading Company, from Accolade Software, is a science fiction "mini-drama," whose plot and outcome are contingent on the player's relationship with the story's characters. The game features detailed graphic depictions of 30 different characters who interact with the player through conversational text.

The story revolves around the PSI-5 Trading Company, a space freighter setting off on a mission to save the inhabitants of the Parvin Frontier from alien invaders. As captain of the ship, you must choose a crew of 5 from 30 applicants, each possessing special skills and a unique personality. The success of the mission hinges on the confidence you have in your crew to handle its responsibilities.

Suggested retail price for the Commodore 64 version is $29.95. The Apple version retails for $34.95.

Accolade Software, 20863 Stevens Creek Blvd., Cupertino, CA 95014.

Circle Reader Service Number 212.

Epyx Games Available For Amiga And Atari ST

Epyx has announced that two of its most popular computer games, and a microcomputer version of a classic mainframe adventure game, will be available for the Amiga and Atari ST this spring. In *Winter Games*, up to eight people can compete in seven events from the Winter Olympics. The original Commodore version of the game featured excellent graphics and sound. *Rogue* was originally a mainframe computer adventure game often played on college campuses. And the *Temple of Apshai Trilogy* offers a wide range of multiple dungeon levels, featuring 1400 separate chambers, plus enhanced high-resolution graphics.

Both versions of all three games are expected to retail for between $19–$39.

Epyx, Inc., 1043 Kiel Ct., Sunnyvale, CA 94089.

Circle Reader Service Number 213.

Bantam Software Promotions

Bantam Electronic Publishing is offering software promotional deals for purchasers of *Sherlock Holmes In "Another Bow," The Fourth Protocol,* and *The Complete Scarsdale Medical Diet.* Through April 15, special rebate coupons can be used to take $5 off the price of each of those programs. And, through March 31, Bantam will take entries in its Mystery Weekend contest, the winner of which will get a weekend for two in Boston to participate in a "mystery weekend" at the famous Parker House hotel. Special Holmes mystery pamphlets are available in many participating software stores. The pamphlets contain a mystery which you solve, and then submit to Bantam for a drawing in mid-April.

Bantam Electronic Publishing, Bantam Books, 666 Fifth Ave., New York, NY 10103.

Circle Reader Service Number 214.

Sports Tutorials Designed By The Pros

Avant-Garde has enlisted the help of three famous professional athletes in developing a line of sports tutorials. *Joe Theismann's Pro Football* offers advice on training and strategy to help develop quarterbacking techniques; improves overall football skills; and helps you understand the finer points of the game. Plays are illustrated through live-action diagrams. *Dave Winfield's Batter Up!* advises on pitchers, batting stance, swing height, grip and hitting strategy to help you develop expert batting techniques. The package also includes Winfield's book, *Batter Up! The Act of Hitting,* and a four-player batting game, *Slugfest!. Chris Evert-Lloyd Tennis* provides animated demonstrations of grip, stroke, game strategy, and specialized exercises. The program helps you learn the rules and choose the best equipment, and teaches concentration techniques to prepare you mentally for a match.

The Commodore 64 version of each program retails for $34.95. The Apple II version (64K RAM minimum) and IBM-PC/PCjr version (128K RAM minimum) retail for $39.95.

Avant-Garde, 37B Commercial Blvd., Novato, CA 94947.

Circle Reader Service Number 215.

Infocom At The Big Top

In *Ballyhoo*, Infocom's new interactive mystery, you are a small-town circus-goer who sticks around after the show to explore the exotic back lot. What you discover is a mysterious underworld of crime and corruption, into which the circus owner's daughter has been kidnapped. In order to find her, you must solve a series of puzzles that are hidden among the circus folk.

Ballyhoo, one of Infocom's standard-level, all-text adventure games, is available for the Apple II-series and Macintosh; Atari XL/XE and ST series, Commodore 64/128 and Amiga, and the IBM PC and PCjr, for a list price of $39.95.

Infocom, Inc., 125 Cambridge Park Dr., Cambridge, MA 02140.

Circle Reader Service Number 216.

New From Better Working

Word Processor with Spellchecker is the third product to be released in Spinnaker's Better Working line of home productivity software. The program is a full-function word processor, with a 50,000-word American Heritage Dictionary to catch spelling mistakes. It also features a 750-word personalized user dictionary, preview mode, microcommands for alternative print styles, and window-based menus and help screens.

The other titles in the Better Working series are *Spreadsheet* and *File and Report. Word Processor with Spellchecker* can perform mailmerge with *Better Working File and Report.* Each program is available for the Apple II series ($59.95) and the Commodore 64/128 ($49.95).

Better Working, Spinnaker, One Kendall Square, Cambridge, MA 02139.

Circle Reader Service Number 217.

©

Classified

SOFTWARE

TI-99/4A Software/Hardware bargains. Hard-to-find items. Huge selection Fast service. Free catalog. D.E.C., Box 690, Hicksville, NY 11801

LOTTO PICKER. Improve your chances for those Million Dollar Jackpots! Picks LOTTO, WIN-4, & Daily Numbers. All USA & Can. games incl. Expandable! IBM/C64/TI99 $29.95. Order Now! 1-800-341-1950 Ext. 77. Mail Orders: Ridge, 170 Broadway, #201-C, NYC, NY 10038. Catalog

PROJECT PLANNING/MANAGEMENT using the C64, SX, or C128. Data sheet for SASE - Prgm for $106.95 (CA res. add 6% sls tx). LAWCO, Dept. C, Box 2009, Manteca, CA 95336

Ganealogy Program for the C64. **"FAMILY TREE"** will produce Pedigree Charts, Family Group Records, Individual Files, Indexes, Searches of Ancestors. LDS version available. "The Best" genealogy program for the 64. $49.95, GENEALOGY SOFTWARE, POB 1151, PORT HURON, MI 48061, (519) 344-3990.

Animal Records maintained with **"PETIGREE"** for the C64. Produces Litter, Awards, Breeding Show, Individual Records, Pedigree Charts. $69.95. GENEALOGY SOFTWARE, POB 1151, PORT HURON, MI 48061, (519) 344-3990.

FREE SOFTWARE CATALOG!
Call Toll-Free 1-800-554-1162, Tevex, Inc. Save ⅓ off retail prices. We carry SSI, Elect. Arts, Infocom, and many more!

COMMODORE: TRY BEFORE YOU BUY. Top 25 best-selling games, utilities, new releases. Visa, MasterCard. Free brochure. Rent-A-Disk, 908 9th Ave., Huntington, WV 25701 (304) 522-1665

DISCOUNT SOFTWARE. Amiga/Apple/Atari/C64-128/IBM PC-PCjr/TRS-80/Timex/Sinclair. Free Catalog: WMJ DATA SYSTEMS, 4 Butterfly Dr., Hauppauge, NY 11788

3 GAMES-PHOBOS-Destroy a falling moon. 10101 - Play 21 against your computer. STALACTITE ATTACK-Try to survive in a cave with attacking stalactites-requires joystick. ALL THREE for 32K ATARI, ON DISK, $5 check or M.O. to: M. A. Grossman, W205 N6844 Woodward Ct., Menomonee Falls, WI 53051

Dr. T's MUSIC SOFTWARE
A music composition system with true word processing capabilities for APPLE and COMMODORE computers. Also algorithmic composition programs for computer generation of sequences. Fully editable Bach, Keyboard, Bass and Drum disks. Patch librarians available for Yamaha and Casio synthesizers. Reviewed in Jan '85 COMPUTE! (617) 244-6954, Dr. T's, Dept. C, 66 LOUISE RD., CHESTNUT HILL, MA 02167

DATA BASE 130 FOR THE ATARI 130 XE
Data files up to 64K. Sort on any field. Print address labels or listings. $25 Bill Von Sennet, P.O. Box 14585, Pittsburgh, PA 15234. Specify 810 or 1050

IBM PCjr OWNERS: We have the software you want. We also have complete 2nd drive additions or kits. FREE CATALOG. OOWL SOFTWARE, 1435 Burnley Sq. N., Columbus, OH 43229

FREE SOFTWARE for C64, C128, IBM & CPM. For info send large stamped (39¢) return envelope to: PUBLIC DOMAIN USERS GROUP, PO Box 1442-A1, Orange Park, FL 32067

TI-99/4A QUALITY SOFTWARE for Business, Home and Entertainment * * BONUS Software Offer! * * Send for FREE catalog to MICRO - BIZ HAWAII, BOX 1108, Pearl City, HI 96782

TAX SPREADSHEET FOR C64, ATARI, TI: $19.95. + $2 s/h. Includes 1040, 2106, 2441, A,B,C,D,E,G,SE,W,1040A. Yearly updates $10. TI-99/4A w/ExtBasic, Atari w/48K or more. Specify disk or tape. Steve Karasek, 855 Diversey, St. Louis, MO 63126 (314) 961-2052

ATTENTION TI99/4A OWNERS !!!
See our ad in Product Mart.
The 99/4A National Assistance Group
(305) 583-0467

**** ATARI USERS ** QUEST FOR LIFE!!**
Realistic Adventure with weapons, killers and traps. $14.95 tape/disk. FREE CATALOG Octavian Software, Box 174, Republic, OH 44867

GAMES FOR C-64! . ANY CHOICE $19.95 T/D. Navy War, Startrek, Football, Risk. Free game w/each buy. RK Games, 420 W. Central, Medicine Lodge, KS 67104 (316) 886-5097

MISCELLANEOUS

HELP IS ON THE WAY!
Just call 1-800-334-0868 to get your free copy of the latest COMPUTE! Books Catalog! If you need help in getting information on all of the latest COMPUTE! book titles available plus all COMPUTE! backlist titles, call us today!

Maxell MD1, $1.29-MD2, $1.99. Dysan 104/1D, $1.79-104/2D, $2.39. Shipping $3.75. Also Verbatim, IBM, 3M, BASF. TAPE WORLD, 220 Spring St., Butler, PA 16001, 1-800-245-6000. Visa, MC.

EARN MONEY, PART OR FULL TIME, AT HOME with your computer-manual & forms-$9.95. Write Computer Programs for Profit! How-to guide with forms, letters, tips-$7.95. Also-Computer Consultant Handbook. How to earn $ consulting with business-$7.95. JV Tech, P.O. Box 563, Ludington, MI 49431

Discount computer printer ribbons for all makes/models Ex: Epson 1500 Nylon $6.99 Catalog. TWS 1314 4th Ave., Coraopolis, PA 15108 (412) 262-1482 Visa or MasterCard

NEED THAT CIRCUIT DESIGNED into artwork, or that artwork fabbed into circuit boards, and the PCB's assembled? Then send requirements for free quote to the PC Master, 10221 Slater Ave., Suite 103-329, Fountain Valley, CA 92708

1985 INDEX TO ALL COMMODORE MAGAZINES. C64/128 disk with search/browse/print of articles, programs, reviews. Send $16.95 ck or m.o. to: PCdex, Box 563, Dayton, OH 45409

IBM PCjr REPORT: THE NATIONAL NEWSLETTER. PCjr-specific articles, reviews, Public Domain from across the nation. $18./yr. PCjr CLUB, POB 95067, Schaumberg, IL 60195

RIBBONS for MOST PRINTERS at LOW PRICES!
DELTA MICRONICS
BOX 10933, ERIE, PA 16514-0933
(814) 455-5667

ROBOT KITS - The MOVIT family of computer robots that you have been reading about.
PEPPY - navigates by two-way sound sensor, zips around on three wheels, $24.99 + $2 s/h.
AVOIDER - infra-red obstacle avoidance system,, walks on six legs, $44.99 + $2 s/h.
MR. BOOTSMAN - Wired remote control six leg robot. Two motor control, $34.99 + $2 s/h.
Catalog $1.00. Send orders to: CompuTek, 530 South 6th St., Sunnyside, WA 98944

COMPUTE! Classified is a low-cost way to tell over 350,000 microcomputer owners about your product or service.

Rates: $25 per line, minimum of four lines. Any or all of the first line set in capital letters at no charge. Add $15 per line for boldface words, or $50 for the entire ad set in boldface (any number of lines.)

Terms: Prepayment is required. Check, money order, American Express, Visa, or MasterCard is accepted. Make checks payable to COMPUTE! Publications

Form: Ads are subject to publisher's approval and must be either typed or legibly printed. One line equals 40 letters and spaces between words. Please underline words to be set in boldface

General Information: Advertisers using post office box numbers in their ads must supply permanent address and telephone numbers. Orders will not be acknowledged. Ad will appear in next available issue after receipt

Closing: 10th of the third month preceding cover date (e.g., June issue closes March 10th). Send order and remittance to: Harry Blair, Classified Manager, COMPUTE!, P.O. Box 5406, Greensboro, NC 27403. To place an ad by phone, call Harry Blair at (919) 275-9809

Notice: COMPUTE! Publications cannot be responsible for offers or claims of advertisers, but will attempt to screen out misleading or questionable copy

MLX Machine Language Entry Program For Atari

Charles Brannon, Program Editor

MLX is a labor-saving utility that allows almost fail-safe entry of machine language programs published in COMPUTE!. *You need to know nothing about machine language to use MLX—it was designed for everyone.*

"MLX" is a new way to enter long machine language (ML) programs with a minimum of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter illegal characters when you should be typing numbers. It won't let you enter numbers greater than 255 (forbidden in ML). It won't let you enter the wrong numbers on the wrong line. In addition, MLX creates a ready-to-use tape or disk file.

Using MLX

Type in and save MLX (you'll want to use it in the future). When you're ready to type in an ML program, run MLX. MLX asks you for three numbers: the starting address, the ending address, and the run/init address. These numbers are given in the article accompanying the ML program presented in MLX format. You must also choose one of three options for saving the file: as a boot tape, as disk binary file, or as boot disk. The article with the ML program should specify which formats may be used.

When you run MLX, you'll see a prompt corresponding to the starting address. The prompt is the current line you are entering from the listing. It increases by six each time you enter a line. That's because each line has seven numbers—six actual data numbers plus a checksum number. The checksum verifies that you typed the previous six numbers correctly. If you enter any of the six numbers wrong, or enter the checksum wrong, the computer rings a buzzer and prompts you to reenter the line. If you enter it correctly, a bell tone sounds and you continue to the next line.

MLX accepts only numbers as input. If you make a typing error, press the DEL/BACK SPACE; the entire number is deleted. You can press it as many times as necessary back to the start of the line. If you enter three-digit numbers as listed, the computer automatically prints the comma and goes on to accept the next number. If you enter fewer than three digits, you can press the comma key, the space bar, or the RETURN key to advance to the next number. The checksum automatically appears in inverse video for emphasis.

MLX Commands

When you finish typing an ML listing (assuming you type it all in one session), you can then save the completed program on tape or disk. Follow the screen instructions. If you get any errors while saving, you probably have a bad disk, or the disk is full, or you've made a typo when entering the MLX program itself.

You don't have to enter the whole ML program in one sitting. MLX lets you enter as much as you want, save it, and then reload the file from tape or disk later. MLX recognizes these commands:

CTRL-S	Save
CTRL-L	Load
CTRL-N	New Address
CTRL-D	Display

To issue a command, hold down the CTRL key (CONTROL on the XL models) and press the indicated key. When you enter a command, MLX jumps out of the line you've been typing, so we recommend you do it at a new prompt. Use the Save command (CTRL-S) to save what you've been working on. It will save on tape or disk, as if you've finished, but the tape or disk won't work, of course, until you finish the typing. Remember to make a note of what address you stop at. The next time you run MLX, answer all the prompts as you did before—regardless of where you stopped typing—then insert the disk or tape. When you get to the line number prompt, press CTRL-L to reload the partly completed file into memory. Then use the New Address command to resume typing.

To use the New Address command, press CTRL-N and enter the address where you previously stopped. The prompt will change, and you can then continue typing. Always enter a New Address that matches up with one of the line numbers in the MLX-format listing, or else the checksum won't work. The Display command lets you display a section of your typing. After you press CTRL-D, enter two addresses within the line number ran_e of the listing. You can break out of the listing display and return to the prompt by pressing any key.

Atari MLX: Machine Language Entry

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" in this issue of COMPUTE!

```
DA 100 GRAPHICS Ø:DL=PEEK(S6Ø
       )+2S6*PEEK(S61)+4:POKE
        DL-1,71:POKE DL+2,6
NJ 11Ø POSITION B,Ø:? "MLX":P
       OSITION 23,Ø:? "[illegible]
       [illegible]":POKE 71Ø,Ø:?
JK 12Ø ? "Starting Address";:
       INPUT BEG:? "  Ending
       Address";:INPUT FIN:?
       "Run/Init Address";:IN
       PUT STARTAOR
DD 13Ø OIM A(6),BUFFER$(FIN-B
       EG+127),T$(2Ø),F$(2Ø),
       CIO$(7),SECTOR$(12B),O
       SKINV$(6)
JJ 14Ø OPEN #1,4,Ø,"K:":? :?
       ,"Tape or Disk:";
BM 15Ø BUFFER$=CHR$(Ø):BUFFER
       $(FIN-BEG+3Ø)=BUFFER$:
       BUFFER$(2)=BUFFER$:SEC
       TOR$=BUFFER$
BC 16Ø AOOR=BEG:CIO$="hhh":CI
       O$(4)=CHR$(17Ø):CIO$(5
       )="LV":CIO$(7)=CHR$(22
       B)
EJ 17Ø GET #1,MEOIA:IF MEOIA<
       >B4 ANO MEOIA<>6B THEN
        17Ø
PO 1BØ ? CHR$(MEOIA):? :IF ME
       OIA<>ASC("T") THEN BUF
       FER$="":GOTO 2SØ
PL 19Ø BEG=BEG-24:BUFFER$=CHR
       $(Ø):BUFFER$(2)=CHR$(I
       NT((FIN-BEG+127)/12B))
KF 2ØØ H=INT(BEG/2S6):L=BEG-H
       *256:BUFFER$(3)=CHR$(L
       ):BUFFER$(4)=CHR$(H)
EC 21Ø PINIT=BEG+B:H=INT(PINI
       T/2S6):L=PINIT-H*2S6:B
       UFFER$(5)=CHR$(L):BUFF
       ER$(6)=CHR$(H)
PB 22Ø FOR I=7 TO 24:REAO A:B
       UFFER$(I)=CHR$(A):NEXT
        I:OATA 24,96,169,6Ø,1
       41,2,211,169,Ø,133,1Ø,
       169,Ø,133,11,76,Ø,Ø
DP 23Ø H=INT(STARTAOR/2S6):L=
       STARTAOR-H*2S6:BUFFER$
       (15)=CHR$(L):BUFFER$(1
       9)=CHR$(H)
KL 24Ø BUFFER$(23)=CHR$(L):BU
       FFER$(24)=CHR$(H)
HI 25Ø IF MEOIA<>ASC("O") THE
       N 36Ø
OO 26Ø ? :? "Boot Disk or Bin
       ary File:";
LI 27Ø GET #1,OTYPE:IF OTYPE<
       >6B AND OTYPE<>7Ø THEN
        27Ø
BM 2BØ ? CHR$(OTYPE):IF OTYPE
       =7Ø THEN 36Ø
[illegible] 29Ø BEG=BEG-3Ø:BUFFER$=CHR
       $(Ø):BUFFER$(2)=CHR$(I
        NT((FIN-BEG+127)/128))
KG 300  H=INT(BEG/256):L=BEG-H
        *256:BUFFER$(3)=CHR$(L
        ):BUFFER$(4)=CHR$(H)
HH 310  PINIT=STARTADR:H=INT(P
        INIT/256):L=PINIT-H*25
        6:BUFFER$(5)=CHR$(L):B
        UFFER$(6)=CHR$(H)
AO 320  RESTORE 330:FOR I=7 TD
         30:READ A:BUFFER$(I)=
        CHR$(A):NEXT I
GA 330  DATA 169,Ø,141,231,2,1
        33,14,169,Ø,141,232,2,
        133,15,169,Ø,133,1Ø,16
        9,Ø,133,11,24,96
OB 340  H=INT(BEG/256):L=BEG-H
        *256:BUFFER$(8)=CHR$(L
        ):BUFFER$(15)=CHR$(H)
OO 350  H=INT(STARTADR/256):L=
        STARTAOR-H*256:BUFFER$
        (22)=CHR$(L):8UFFER$(2
        6)=CHR$(H)
JP 360  GRAPHICS Ø:POKE 712,1Ø
        :POKE 71Ø,1Ø:PDKE 7Ø9,
        2
JK 370  ? ADDR;":";:FOR J=1 TD
         6
NF 380  GOSUB 57Ø:IF N=-1 THEN
         J=J-1:GOTD 38Ø
BF 390  IF N=-19 THEN 72Ø
OI 400  IF N=-12 THEN LET REAO
        =1:GOTO 72Ø
AI 410  TRAP 41Ø:IF N=-14 THEN
         ? :? "New Address";:I
        NPUT ADDR:? :BOTD 37Ø
HO 420  TRAP 4ØØØØ:IF N<>-4 TH
        EN 4BØ
AJ 430  TRAP 43Ø:? :? "Display
        :From";:INPUT F:? ,"To
        ";:INPUT T:TRAP 32767
ML 440  IF F<BEG DR F>FIN DR T
        <BEG DR T>FIN OR T<F T
        HEN ? CHR$(253);"At le
        ast ";BEG;", Not More
        Than ";FIN:GOID 43Ø
MH 450  FOR I=F TO T STEP 6:?
        :? I;":";:FDR K=Ø TO 5
        :N=PEEK(AOR(BUFFER$)+I
        +K-BEG):T$="ØØØ":T$(4-
        LEN(STR$(N)))=STR$(N)
MA 460  IF PEEK(764)<255 THEN
        GET #1,A:PDP :PDP :? :
        GOTD 37Ø
FM 470  ? T$;",";:NEXT K:? CHR
        $(126);:NEXT I:? :? :G
        DTD 37Ø
GA 480  IF N<Ø THEN ? :GOTO 37
        Ø
MH 490  A(J)=N:NEXT J
JM 500  CKSUM=ADDR-INT(ADDR/25
        6)*256:FOR I=1 TO 6:CK
        SUM=CKSUM+A(I):CKSUM=C
        KSUM-256*(CK5UM>255):N
        EXT I
KK 510  RF=128:SOUNO Ø,2ØØ,12,
        8:GOSU8 57Ø:SOUND Ø,Ø,
        Ø,Ø:RF=Ø:? CHR$(126)
CH 520  IF N<>CK5UM THEN ? :?
        "Incorrect";CHR$(253);
        :? :GDTD 37Ø
EK 530  FDR W=15 TD Ø STEP -1:
        SOUNO Ø,5Ø,1Ø,W:NEXT W
FL 540  FDR I=1 TD 6:POKE ADR(
        BUFFER$)+ADDR-BEG+I-1,
        A(I):NEXT I
HB 550  ADOR=AOOR+6:IF AODR<=F
        IN THEN 37Ø
GH 560  GOTO 71Ø
FI 570  N=Ø:Z=Ø
PH 5BØ  GET #1,A:IF A=155 OR A
        =44 OR A=32 THEN 67Ø
FB 590  IF A<32 THEN N=-A:RETU
        RN
EB 600  IF A<>126 THEN 63Ø
ML 610  GDSUB 69Ø:IF I=1 AND T
        =44 THEN N=-1:? CHR$(1
        26);:GOTO 69Ø
GH 620  GOTO 57Ø
GJ 630  IF A<48 DR A>57 THEN 5
        8Ø
AN 640  ? CHR$(A+RF);:N=N*1Ø+A
        -48
EB 650  IF N>255 THEN ? CHR$(2
        53);:A=126:GOTO 6ØØ
EH 660  Z=Z+1:IF Z<3 THEN 58Ø
JH 670  IF Z=Ø THEN ? CHR$(253
        );:GOTD 57Ø
KC 6BØ  ? ",";:RETURN
HO 690  PDKE 752,1:FOR I=1 TD
        3:? CHR$(3Ø);:GET #6,T
        :IF T<>44 AND T<>58 TH
        EN ? CHR$(A);:NEXT I
PI 700  PDKE 752,Ø:? " ";CHR$(
        126);:RETURN
KM 710  GRAPHICS Ø:POKE 71Ø,26
        :POKE 712,26:POKE 7Ø9,
        2
FF 720  IF MEOIA=ASC("T") THEN
         89Ø
OJ 730  REM DISK
OK 740  IF READ THEN ? :? "Loa
        d File":?
IG 750  IF OTYPE<>7Ø THEN 1Ø4Ø
AE 760  ? :? "Enter AUTORUN.SY
        S for automatic use":?
         :? "Enter filename":I
        NPUT T$
BF 770  F$=T$:IF LEN(T$)>2 THE
        N IF T$(1,2)<>"D:" THE
        N F$="D:":F$(3)=T$
HJ 780  TRAP 87Ø:CLOSE #2:OPEN
         #2,8-4*READ,Ø,F$:? :?
         "Working..."
JH 790  IF READ THEN FOR I=1 T
        O 6:BET #2,A:NEXT I:GO
        TO 82Ø
PD BØØ  PUT #2,255:PUT #2,255
OJ 810  H=INT(BEB/256):L=8EB-H
        *256:PUT #2,L:PUT #2,H
        :H=INT(FIN/256):L=FIN-
        H*256:PUT #2,L:PUT #2,
        H
NF B20  8DSU8 97Ø:IF PEEK(195)
        >1 THEN 87Ø
IF 830  IF STARTADR=Ø OR READ
        THEN B5Ø
FO 840  PUT #2,224:PUT #2,2:PU
        T #2,225:PUT #2,2:H=IN
        T(STARTADR/256):L=STAR
        TAOR-H*256:PUT #2,L:PU
        T #2,H
BC 850  TRAP 4ØØØØ:CLOSE #2:?
        "Finished.":IF READ TH
        EN ? :? :LET READ=Ø:GO
        TO 36Ø
HF 860  END
FO 870  ? "Error ";PEEK(195);"
         trying to access":? F
        $:CLDSE #2:? :GOTO 76Ø
NC 880  REM BOOT TAPE
HH 890  IF READ THEN ? :? "Rea
        d Tape"
HI 900  ? :? :? "Insert, Rewin
        d Tape.":? "Press PLAY
         ";:IF  NDT READ THEN
        ? "& RECORO"
LP 910  ? :? "Press RETURN whe
        n ready:";
JH 920  TRAP 96Ø:CLOSE #2:DPEN
         #2,8-4*READ,12B,"C:":
        ? :? "Working..."
HH 930  GDSUB 97Ø:IF PEEK(195)
        >1 THEN 96Ø
BC 940  CLOSE #2:TRAP 4ØØØØ:?
        "Finished.":? :? :IF R
        EAD THEN LET READ=Ø:GO
        TO 36Ø
HF 950  END
CD 960  ? :? "Error ";PEEK(195
        );" when reading/writi
        ng boot tape":? :CLOSE
         #2:GOTO 89Ø
MB 970  REM CIO Load/Save File
        #2 opened  READ=Ø for
        write, READ=1 for read
EA 980  X=32:REM File#2,$2Ø
EF 990  ICCOM=B34:ICBADR=B36:I
        C8LEN=84Ø:ICSTAT=B35
HD 1000  H=INT(ADR(BUFFER$)/25
         6):L=ADR(BUFFER$)-H*2
         56:POKE ICBADR+X,L:PD
         KE ICBADR+X+1,H
FH 1010  L=FIN-BEG+1:H=INT(L/2
         56):L=L-H*256:POKE IC
         BLEN+X,L:POKE ICBLEN+
         X+1,H
MD 1020  POKE ICCOM+X,11-4*REA
         D:A=U5R(ADR(CIO$),X)
SG 1030  PDKE 195,PEEK(ICSTAT)
         :RETURN
KA 1040  REM SECTOR I/O
GC 1050  IF READ THEN 11ØØ
HE 1060  ? :? "Format Disk In
         Drive 1? (Y/N):";
FC 1070  GET #1,A:IF A<>78 ANO
          A<>89 THEN 1Ø7Ø
EC 1080  ? CHR$(A):IF A=7B THE
         N 11ØØ
CP 1090  ? :? "Formatting...":
         XIO 254,#2,Ø,Ø,"D:":?
          "Format Complete":?
AC 1100  NR=INT((FIN-8EG+127)/
         12B):BUFFER$(FIN-BEG+
         2)=CHR$(Ø):IF READ TH
         EN ? "Reading...":GOT
         D 112Ø
LE 1110  ? "Writing..."
LI 1120  FOR I=1 TO NR:5=I
IO 1130  IF READ THEN GOSUB 12
         2Ø:BUFFER$(I*128-127)
         =5ECTOR$:GOTO 116Ø
PL 1140  SECTDR$=BUFFER$(I*12B
         -127)
AM 1150  GDSUB 122Ø
DH 1160  IF PEEK(DSTATS)<>1 TH
         EN 12ØØ
FB 1170  NEXT I
GN 1180  IF  NOT READ THEN END
OH 1190  ? :? :LET READ=Ø:GOTO
          36Ø
JJ 1200  ? "Error on disk acce
         ss.":? "May need form
         atting.":GDTD 1Ø4Ø
KI 1210  REM
BL 1220  REM SECTOR ACCESS SU
         BROUTINE
IB 1230  REM Drive ONE
IH 1240  REM Pass buffer in SE
         CTOR$
MP 1250  REM sector # in varia
         ble S
EG 1260  REM READ=1 for read,
KJ 1270  REM READ=Ø for write
BK 12BØ  BASE=3*256
GL 1290  DUNIT=BASE+1:DCDMND=B
         ASE+2:OSTATS=BASE+3
NL 1300  DBUFLO=BASE+4:DBUFHI=
         BASE+5
AI 1310  DBYTLO=BASE+B:DBYTHI=
         BASE+9
JA 1320  DAUX1=8ASE+1Ø:DAUX2=B
         ASE+11
PH 1330  REM DIM D5KINV$(4)
CA 1340  DSKINV$="hLS":DSKINV$
         (4)=CHR$(228)
PF 1350  POKE DUNIT,1:A=ADR(5E
         CTDR$):H=INT(A/256):L
         =A-256*H
SP 1360  POKE OBUFHI,H
CO 1370  POKE DBUFLO,L
PD 1380  PDKE DCOMND,B7-5*READ
AA 1390  PDKE OAUX2,INT(S/256)
         :PDKE DAUX1,5-PEEK(DA
         UX2)*256
KJ 1400  A=U5R(ADR(O5KINV$))
KS 1410  RETURN                ©
```

MLX Machine Language Entry Program For Commodore 64

Charles Brannon, Program Editor

MLX is a labor-saving utility that allows almost fail-safe entry of machine language programs published in COMPUTE!. *You need to know nothing about machine language to use MLX—it was designed for everyone. At least 8K expansion memory is required.*

MLX is a new way to enter long machine language (ML) programs with a minimum of fuss. MLX lets you enter the numbers from a special list that looks similar to BASIC DATA statements. It checks your typing on a line-by-line basis. It won't let you enter illegal characters when you should be typing numbers. It won't let you enter numbers greater than 255 (forbidden in ML). It won't let you enter the wrong numbers on the wrong line. In addition, MLX creates a ready-to-use tape or disk file.

Using MLX

Type in and save the appropriate version of MLX (you'll want to use it in the future). When you're ready to type in an ML program, run MLX. MLX for the 64 asks you for two numbers: the starting address and the ending address. These numbers are given in the article accompanying the ML program.

When you run MLX, you'll see a prompt corresponding to the starting address. The prompt is the current line you are entering from the listing. It increases by six each time you enter a line. That's because each line has seven numbers—six actual data numbers plus a *checksum number*. The checksum verifies that you typed the previous six numbers correctly. If you enter any of the six numbers wrong, or enter the checksum wrong, the computer rings a buzzer and prompts you to reenter the line. If you enter it correctly, a bell tone sounds and you continue to the next line.

MLX accepts only numbers as input. If you make a typing error, press the INST/DEL key; the entire number is deleted. You can press it as many times as necessary back to the start of the line. If you enter three-digit numbers as listed, the computer automatically prints the comma and goes on to accept the next number. If you enter less than three digits, you can press either the space bar or RETURN key to advance to the next number. The checksum automatically appears in inverse video for emphasis.

To simplify your typing, MLX redefines part of the keyboard as a numeric keypad (lines 581–584):

	U	I	O			7	8	9
H	J	K	L	become	0	4	5	6
	M	,	.			1	2	3

64 MLX Commands

When you finish typing an ML listing (assuming you type it all in one session), you can then save the completed program on tape or disk. Follow the screen instructions. If you get any errors while saving, you probably have a bad disk, or the disk is full, or you've made a typo when entering the MLX program itself.

You don't have to enter the whole ML program in one sitting. MLX lets you enter as much as you want, save it, and then reload the file from tape or disk later. MLX recognizes these commands:

SHIFT-S: Save
SHIFT-L: Load
SHIFT-N: New Address
SHIFT-D: Display

When you enter a command, MLX jumps out of the line you've been typing, so we recommend you do it at a new prompt. Use the Save command to save what you've been working on. It will save on tape or disk, as if you've finished, but the tape or disk won't work, of course, until you finish the typing. Remember what address you stop at. The next time you run MLX, answer all the prompts as you did before, then insert the disk or tape. When you get to the entry prompt, press SHIFT-L to reload the partly completed file into memory. Then use the New Address command to resume typing.

To use the New Address command, press SHIFT-N and enter the address where you previously stopped. The prompt will change, and you can then continue typing. Always enter a New Address that matches up with one of the line numbers in the special listing, or else the checksum won't work. The Display command lets you display a section of your typing. After you press SHIFT-D, enter two addresses within the line number range of the listing. You can abort the listing by pressing any key.

64 MLX: Machine Language Entry

```
10 REM LINES CHANGED FROM MLX
   {SPACE}VERSION 2.00 ARE 750
   ,765,770 AND 860    :rem 50
20 REM LINE CHANGED FROM MLX V
   ERSION 2.01 IS 300 :rem 147
100 PRINT"{CLR}⁅6⁆";CHR$(142);
    CHR$(8);:POKE53281,1:POKE5
    3280,1               :rem 67
101 POKE 788,52:REM DISABLE RU
    N/STOP              :rem 119
110 PRINT"{RVS}{39 SPACES}";
                        :rem 176
120 PRINT"{RVS}{14 SPACES}
    {RIGHT}{OFF}⁅*⁆£{RVS}
    {RIGHT} {RIGHT}{2 SPACES}
    ⁅*⁆{OFF}⁅*⁆£{RVS}£{RVS}
    {14 SPACES}";       :rem 250
130 PRINT"{RVS}{14 SPACES}
    {RIGHT} ⁅G⁆{RIGHT}
    {2 RIGHT} {OFF}£{RVS}£
    ⁅*⁆{OFF}⁅*⁆{RVS}
    {14 SPACES}";        :rem 35
140 PRINT"{RVS}{41 SPACES}"
                        :rem 120
200 PRINT"{2 DOWN}{PUR}{BLK} M
    ACHINE LANGUAGE EDITOR VER
    SION 2.02{5 DOWN}":rem 238
210 PRINT"⁅5⁆{2 UP}STARTING AD
    DRESS?{8 SPACES}{9 LEFT}";
                        :rem 143
215 INPUTS:F=1-F:C$=CHR$(31+11
    9*F)                :rem 166
220 IFS<256OR(S>40960ANDS<4915
    2)ORS>53247THENGOSUB3000:G
    OTO210              :rem 235
225 PRINT:PRINT:PRINT :rem 180
230 PRINT"⁅5⁆{2 UP}ENDING ADDR
    ESS?{8 SPACES}{9 LEFT}";:I
    NPUTE:F=1-F:C$=CHR$(31+119
    *F)                  :rem 20
240 IFE<256OR(E>40960ANDE<4915
    2)ORE>53247THENGOSUB3000:G
    OTO230              :rem 1B3
250 IFE<STHENPRINTC$;"{RVS}END
    ING < START{2 SPACES}":GOS
    UB1000:GOTO 230     :rem 176
260 PRINT:PRINT:PRINT :rem 179
300 PRINT"{CLR}";CHR$(14):AD=S
                         :rem 56
310 A=1:PRINTRIGHT$("0000"+MID
    $(STR$(AD),2),5);":";
                         :rem 33
315 FORJ=ATO6            :rem 33
320 GOSUB570:IFN=-1THENJ=J+N:G
    OTO320              :rem 228
390 IFN=-211THEN 710     :rem 62
400 IFN=-204THEN 790     :rem 64
410 IFN=-206THENPRINT:INPUT"
    {DOWN}ENTER NEW ADDRESS";Z
    Z                    :rem 44
415 IFN=-206THENIFZZ<SORZZ>ETH
    ENPRINT"{RVS}OUT OF RANGE"
    :GOSUB1000:GOTO410:rem 225
417 IFN=-206THENAD=ZZ:PRINT:GO
    TO310               :rem 23B
420 IF N<>-196 THEN 480
                        :rem 133
430 PRINT:INPUT"DISPLAY:FROM";
    F:PRINT,"TO";:INPUTT
                        :rem 234
440 IFF<SORF>EORT<SORT>ETHENPR
    INT"AT LEAST";S;"{LEFT}, N
    OT MORE THAN";E:GOTO430
                        :rem 159
450 FORI=FTOTSTEP6:PRINT:PRINT
    RIGHT$("0000"+MID$(STR$(I)
    ,2),5);":";          :rem 30
451 FORK=0TO5:N=PEEK(I+K):PRIN
    TRIGHT$("00"+MID$(STR$(N),
    2),3);",";           :rem 66
```

```
460 GETA$:IFA$>""THENPRINT:PRI
    NT:GOTO31Ø              :rem 25
470 NEXTK:PRINTCHR$(2Ø);:NEXTI
    :PRINT:PRINT:GOTO31Ø
                            :rem 5Ø
48Ø IFN<Ø THEN PRINT:GOTO31Ø
                           :rem 168
49Ø A(J)=N:NEXTJ           :rem 199
5ØØ CKSUM=AD-INT(AD/256)*256:F
    ORI=1TO6:CKSUM=(CKSUM+A(I)
    )AND255:NEXT           :rem 2ØØ
51Ø PRINTCHR$(18);:GOSU857Ø:PR
    INTCHR$(146);           :rem 94
511 IFN=-1THENA=6:GOTO315
                           :rem 254
515 PRINTCHR$(2Ø):IFN=CKSUMTHE
    N53Ø                   :rem 122
52Ø PRINT:PRINT"LINE ENTERED W
    RONG : RE-ENTER":PRINT:GOS
    U81ØØØ:GOTO31Ø         :rem 176
53Ø GOSUB2ØØØ              :rem 218
54Ø FORI=1TO6:POKEAD+I-1,A(I):
    NEXT:POKE54272,Ø:POKE54273
    ,Ø                     :rem 227
55Ø AD=AD+6:IF AD<E THEN 31Ø
                           :rem 212
56Ø GOTO 71Ø               :rem 1Ø8
57Ø N=Ø:Z=Ø                 :rem 88
58Ø PRINT"{£}";            :rem 81
581 GETA$:IFA$=""THEN581
                            :rem 95
582 AV=-(A$="M")-2*(A$=",")-3*
    (A$=".")-4*(A$="J")-5*(A$=
    "K")-6*(A$="L")         :rem 41
583 AV=AV-7*(A$="U")-8*(A$="I"
    )-9*(A$="O"):IFA$="H"THENA
    $="Ø"                  :rem 134
584 IFAV>ØTHENA$=CHR$(48+AV)
                           :rem 134
585 PRINTCHR$(2Ø);:A=ASC(A$):I
    FA=13ORA=44ORA=32THEN67Ø
                           :rem 229
59Ø IFA>128THENN=-A:RETURN
                           :rem 137
6ØØ IFA<>2Ø THEN 63Ø        :rem 1Ø
61Ø GOSU869Ø:IFI=1ANDT=44THENN
    =-1:PRINT"{OFF}{LEFT}
    {LEFT}";:GOTO69Ø        :rem 62
62Ø GOTO57Ø                :rem 1Ø9
63Ø IFA<48ORA>57THEN58Ø
                           :rem 1Ø5
64Ø PRINTA$;:N=N*1Ø+A-48
                           :rem 1Ø6
65Ø IFN>255 THEN A=2Ø:GOSU81ØØ
    Ø:GOTO6ØØ              :rem 229
66Ø Z=Z+1:IFZ<3THEN58Ø :rem 71
67Ø IFZ=ØTHENGOSUB1ØØØ:GOTO57Ø
                           :rem 114
68Ø PRINT",";:RETURN       :rem 24Ø
69Ø S%=PEEK(2Ø9)+256*PEEK(21Ø)
    +PEEK(211)             :rem 149
691 FORI=1TO3:T=PEEK(S%-I)
                            :rem 67
695 IFT<>44ANDT<>58THENPOKES%-
    I,32:NEXT              :rem 2Ø5
7ØØ PRINTLEFT$("{3 LEFT}",I-1)
    ;:RETURN                 :rem 7
71Ø PRINT"{CLR}{RVS}*** SAVE *
    **{3 DOWN}"            :rem 236
715 PRINT"{2 DOWN}(PRESS {RVS}
    RETURN{OFF} ALONE TO CANCE
    L SAVE){DOWN}"         :rem 1Ø6
72Ø F$="":INPUT"{DOWN} FILENAM
    E";F$:IFF$=""THENPRINT:PRI
    NT:GOTO31Ø              :rem 71
73Ø PRINT:PRINT"{2 DOWN}{RVS}T
    {OFF}APE OR {RVS}D{OFF}ISK
    : (T/D)"               :rem 228
74Ø GETA$:IFA$<>"T"ANDA$<>"D"T
    HEN74Ø                  :rem 36
75Ø DV=1-7*(A$="D"):IFDV=8THEN
    F$="Ø:"+F$:OPEN15,8,15,"S"
    +F$:CLOSE15            :rem 212
76Ø T$=F$:ZK=PEEK(53)+256*PEEK
    (54)-LEN(T$):POKE782,ZK/25
    6                        :rem 3
762 POKE781,ZK-PEEK(782)*256:P
    OKE78Ø,LEN(T$):SYS65469
                           :rem 1Ø9
763 POKE78Ø,1:POKE781,DV:POKE7
    82,1:SYS65466           :rem 69
765 K=S:POKE254,K/256:POKE253,
    K-PEEK(254)*256:POKE78Ø,25
    3                       :rem 17
766 K=E+1:POKE782,K/256:POKE78
    1,K-PEEK(782)*256:SYS65496
                           :rem 235
77Ø IF(PEEK(783)AND1)OR(191AND
    ST)THEN78Ø             :rem 111
775 PRINT"{DOWN}DONE.{DOWN}":G
    OTO31Ø                 :rem 113
78Ø PRINT"{DOWN}ERROR ON SAVE.
    {2 SPACES}TRY AGAIN.":IFDV
    =1THEN72Ø              :rem 171
781 OPEN15,8,15:INPUT#15,E1$,E
    2$:PRINTE1$;E2$:CLOSE15:GO
    TO72Ø                  :rem 1Ø3
79Ø PRINT"{CLR}{RVS}*** LOAD *
    **{2 DOWN}"            :rem 212
795 PRINT"{2 DOWN}(PRESS {RVS}
    RETURN{OFF} ALONE TO CANCE
    L LOAD)"                :rem 82
8ØØ F$="":INPUT"{2 DOWN} FILEN
    AME";F$:IFF$=""THENPRINT:G
    OTO31Ø                 :rem 144
81Ø PRINT:PRINT"{2 DOWN}{RVS}T
    {OFF}APE OR {RVS}D{OFF}ISK
    : (T/D)"               :rem 227
82Ø GETA$:IFA$<>"T"ANDA$<>"D"T
    HEN82Ø                  :rem 34
83Ø DV=1-7*(A$="D"):IFDV=8THEN
    F$="Ø:"+F$             :rem 157
84Ø T$=F$:ZK=PEEK(53)+256*PEEK
    (54)-LEN(T$):POKE782,ZK/25
    6                        :rem 2
841 POKE781,ZK-PEEK(782)*256:P
    OKE78Ø,LEN(T$):SYS65469
                           :rem 1Ø7
845 POKE78Ø,1:POKE781,DV:POKE7
    82,1:SYS65466           :rem 7Ø
85Ø POKE78Ø,Ø:SYS65493 :rem 11
86Ø IF(PEEK(783)AND1)OR(191AND
    ST)THEN87Ø             :rem 111
865 PRINT"{DOWN}DONE.":GOTO31Ø
                            :rem 96
87Ø PRINT"{DOWN}ERROR ON LOAD.
    {2 SPACES}TRY AGAIN.{DOWN}
    ":IFDV=1THEN8ØØ        :rem 172
88Ø OPEN15,8,15:INPUT#15,E1$,E
    2$:PRINTE1$;E2$:CLOSE15:GO
    TO8ØØ                  :rem 1Ø2
1ØØØ REM BUZZER            :rem 135
1ØØ1 POKE54296,15:POKE54277,45
     :POKE54278,165        :rem 2Ø7
1ØØ2 POKE54276,33:POKE 54273,6
     :POKE54272,5           :rem 42
1ØØ3 FORT=1TO2ØØ:NEXT:POKE5427
     6,32:POKE54273,Ø:POKE5427
     2,Ø:RETURN            :rem 2Ø2
2ØØØ REM BELL SOUND         :rem 78
2ØØ1 POKE54296,15:POKE54277,Ø:
     POKE54278,247         :rem 152
2ØØ2 POKE 54276,17:POKE54273,4
     Ø:POKE54272,Ø          :rem 86
2ØØ3 FORT=1TO1ØØ:NEXT:POKE5427
     6,16:RETURN            :rem 57
3ØØØ PRINTC$;"{RVS}NOT ZERO PA
     GE OR ROM":GOTO1ØØØ
                            :rem 89
```

©

Save Your Copies of COMPUTE!

Protect your back issues of *COMPUTE!* in durable binders or library cases. Each binder or case is custom-made in flag-blue binding with embossed white lettering. Each holds a year of *COMPUTE!*. Order several and keep your issues of *COMPUTE!* neatly organized for quick reference. (These binders make great gifts, too!)

Binders	**Cases:**
$8.50 each;	$6.95 each;
3 for $24.75;	3 for $20.00;
6 for $48.00	6 for $36.00

(Please add $2.50 per unit for orders outside the U.S.)

Send in your prepaid order with the attached coupon

Mail to: Jesse Jones Industries
P.O Box 5120
Dept. Code COTE
Philadelphia, PA 19141

Please send me ________ COMPUTE! □ cases □ binders.
Enclosed is my check or money order for $ ________. (U.S. funds only.)

Name ____________

Address ____________

City ____________

State ________ Zip ______

Satisfaction guaranteed or money refunded

Please allow 4–6 weeks for delivery.

COMPUTE!'s Guide To Typing In Programs

Computers are precise—type the program *exactly* as listed, including necessary punctuation and symbols, except for special characters noted below. We have provided a special listing convention as well as a program to check your typing—"The Automatic Proofreader."

Programs for the IBM, TI-99/4A, and Atari ST models should be typed exactly as listed; no special characters are used. Programs for Commodore, Apple, and Atari 400/800/XL/XE computers may contain some hard-to-read special characters, so we have a listing system that indicates these control characters. You will find these Commodore and Atari characters in curly braces; *do not type the braces.* For example, {CLEAR} or {CLR} instructs you to insert the symbol which clears the screen on the Atari or Commodore machines. A complete list of these symbols is shown in the tables below. For Commodore, Apple, and Atari, a single symbol by itself within curly braces is usually a control key or graphics key. If you see {A}, hold down the CONTROL key and press A. This will produce a reverse video character on the Commodore (in quote mode), a graphics character on the Atari, and an invisible control character on the Apple.

Graphics characters entered with the Commodore logo key are enclosed in a special bracket: [<A>]. In this case, you would hold down the Commodore logo key as you type A. Our Commodore listings are in uppercase, so shifted symbols are underlined. A graphics heart symbol (SHIFT-S) would be listed as S. One exception is {SHIFT-SPACE}. When you see this, hold down SHIFT and press the space bar. If a number precedes a symbol, such as {5 RIGHT}, {6 S}, or [<8 Q>], you would enter five cursor rights, six shifted S's, or eight Commodore-Q's. On the Atari, inverse characters (white on black) should be entered with the inverse video

Atari 400/800/XL/XE

When you see	Type	See	
{CLEAR}	ESC SHIFT <	↖	Clear Screen
{UP}	ESC CTRL -	↑	Cursor Up
{DOWN}	ESC CTRL =	↓	Cursor Down
{LEFT}	ESC CTRL +	←	Cursor Left
{RIGHT}	ESC CTRL *	→	Cursor Right
{BACK S}	ESC DELETE	◀	Backspace
{DELETE}	ESC CTRL DELETE	▣	Delete character
{INSERT}	ESC CTRL INSERT	▣	Insert character
{DEL LINE}	ESC SHIFT DELETE	▣	Delete line
{INS LINE}	ESC SHIFT INSERT	▣	Insert line
{TAB}	ESC TAB	▶	TAB key
{CLR TAB}	ESC CTRL TAB	▣	Clear tab
{SET TAB}	ESC SHIFT TAB	▣	Set tab stop
{BELL}	ESC CTRL 2	▣	Ring buzzer
{ESC}	ESC ESC	↰	ESCape key

Commodore PET/CBM/VIC/64/128/16/+4

When You Read:	Press:	See:	When You Reed:	Press:	See:
{CLR}	SHIFT CLR/HOME	♥	[< 1 >]	COMMODORE 1	■
{HOME}	CLR/HOME	S	[< 2 >]	COMMODORE 2	◜
{UP}	SHIFT ↑ CRSR ↓	●	[< 3 >]	COMMODORE 3	⊠
{DOWN}	↑ CRSR ↓	Q	[< 4 >]	COMMODORE 4	▣
{LEFT}	SHIFT ← CRSR →	▌	[< s >]	COMMOOORE S	■
{RIGHT}	← CRSR →	]	[< 6 >]	COMMODORE 6	▐
{RVS}	CTRL 9	R	[< 7 >]	COMMOOORE 7	●
{OFF}	CTRL 0	■	[< 8 >]	COMMOOORE 8	▚
{BLK}	CTRL 1	■	{ F1 }	f1	■
{WHT}	CTRL 2	E	{ F2 }	SHIFT f1	■
{RED}	CTRL 3	£	{ F3 }	f3	■
{CYN}	CTRL 4	◤	{ F4 }	SHIFT f3	■
{PUR}	CTRL 5	▒	{ FS }	f5	■
{GRN}	CTRL 6	↑	{ F6 }	SHIFT f5	■
{BLU}	CTRL 7	←	{ F7 }	f7	■
{YEL}	CTRL 8	π	{ F8 }	SHIFT f7	■
			4	←	▒

key (Atari logo key on 400/800 models).

Whenever more than two spaces appear in a row, they are listed in a special format. For example, {6 SPACES} means press the space bar six times. Our Commodore listings never leave a single space at the end of a line, instead moving it to the next printed line as {SPACE}.

Amiga program listings contain only one special character, the left arrow (←) symbol. This character marks the end of each program line. Wherever you see a left arrow, press RETURN or move the cursor off the line to enter that line into memory. Don't try to type in the left arrow symbol; it's there only as a marker to indicate where each program line ends.

The Automatic Proofreader

Type in the appropriate program listed below, then save it for future use. The Commodore Proofreader works on the Commodore 128, 64, Plus/4, 16, and VIC-20. Don't omit any lines, even if they contain unfamiliar commands or you think they don't apply to your computer. When you run the program, it installs a machine language program in memory and erases its BASIC portion automatically (so be sure to save several copies before running the program for the first time). If you're using a Commodore 128, Plus/4 or 16, do *not* use any GRAPHIC commands while the Proofreader is active. You should disable the Commodore Proofreader before running any other program. To do this, either turn the computer off and on or enter SYS 64738 (for the 64), SYS 65341 (128), SYS 64802 (VIC-20), or SYS 65526 (Plus/4 or 16). To reenable the Proofreader, reload the program and run it as usual. Unlike the original VIC/64 Proofreader, this version works the same with disk or tape.

On the Atari, run the Proofreader to activate it (the Proofreader remains active in memory as a machine language program); you must then enter NEW to erase the BASIC loader. Pressing SYSTEM RESET deactivates the Atari Proofreader; enter PRINT USR(1536) to reenable it.

The Apple Proofreader erases the BASIC portion of itself after you run it, leaving only the machine language portion in memory. It works with either DOS 3.3 or ProDOS. Disable the Apple Proofreader by pressing CTRL-RESET before running another BASIC program.

The IBM Proofreader is a BASIC program that simulates the IBM BASIC line editor, letting you enter, edit, list, save, and load programs that you type. Type RUN to activate. Be sure to leave Caps Lock on, except when typing lowercase characters.

Once the Proofreader is active, try typing in a line. As soon as you press RETURN, either a hexadecimal number (on the Apple) or a pair of letters (on the Commodore, Atari, or IBM) appears. The number or pair of letters is called a *checksum*.

Compare the value displayed on the screen by the Proofreader with the checksum printed in the program listing in the magazine. The checksum is given to the left of each line number. Just type in the program a line at a time (without the printed checksum), press RETURN or Enter, and compare the checksums. If they match, go on to the next line. If not, check your typing; you've made a mistake. Because of the checksum method used, do not type abbreviations, such as ? for PRINT. On the Atari and Apple Proofreaders, spaces are not counted as part of the checksum, so be sure you type the right number of spaces between quote marks. The Atari Proofreader does not check to see that you've typed the characters in the right order, so if characters are transposed, the checksum still matches the listing. The Commodore Proofreader catches transposition errors and ignores spaces unless they're enclosed in quotation marks. The IBM Proofreader detects errors in spacing and transposition.

IBM Proofreader Commands

Since the IBM Proofreader replaces the computer's normal BASIC line editor, it has to include many of the direct-mode IBM BASIC commands. The syntax is identical to IBM BASIC. Commands simulated are LIST, LLIST, NEW, FILES, SAVE, and LOAD. When listing your program, press any key (except Ctrl-Break) to stop the listing. If you enter NEW, the Proofreader prompts you to press Y to be especially sure you mean yes.

Two new commands are BASIC and CHECK. BASIC exits the Proofreader back to IBM BASIC, leaving the Proofreader in memory. CHECK works just like LIST, but shows the checksums along with the listing. After you have typed in a program, save it to disk. Then exit the Proofreader with the BASIC command, and load the program as usual (this replaces the Proofreader in memory). You can now run the program, but you may want to resave it to disk. This will shorten it on disk and make it load faster, but it can no longer be edited with the Proofreader. If you want to convert an existing BASIC program to Proofreader format, save it to disk with SAVE *"filename"*,A.

Program 1: Atari Proofreader

By Charles Brannon, Program Editor

```
100 GRAPHICS 0
110 FOR I=1536 TO 1700:REA
    D A:POKE I,A:CK=CK+A:N
    EXT I
120 IF CK<>19072 THEN ? "E
    rror in DATA Statement
    s.  Check Typing.":END

130 A=USR(1536)
140 ? :? "Automatic Proofr
    eader Now Activated."
150 END
160 DATA 104,160,0,185,26,
    3,201,69,240,7
170 DATA 200,200,192,34,20
    8,243,96,200,169,74
180 DATA 153,26,3,200,169,
    6,153,26,3,162
190 DATA 0,189,0,228,157,7
    4,6,232,224,16
200 DATA 208,245,169,93,14
    1,78,6,169,6,141
210 DATA 79,6,24,173,4,228
    ,105,1,141,95
220 DATA 6,173,5,228,105,0
    ,141,96,6,169
230 DATA 0,133,203,96,247,
    238,125,241,93,6
240 DATA 244,241,115,241,1
    24,241,76,205,238
250 DATA 0,0,0,0,0,32,62,2
    46,8,201
260 DATA 155,240,13,201,32
    ,240,7,72,24,101
270 DATA 203,133,203,104,4
    0,96,72,152,72,138
280 DATA 72,160,0,169,128,
    145,88,200,192,40
290 DATA 208,249,165,203,7
    4,74,74,74,24,105
300 DATA 161,160,3,145,88,
    165,203,41,15,24
310 DATA 105,161,200,145,8
    8,169,0,133,203,104
320 DATA 170,104,168,104,4
    0,96
```

Program 2: IBM Proofreader

By Charles Brannon, Program Editor

```
10 'Automatic Proofreader Vers
   ion 3.0 (Lines 205,206 adde
   d/190 deleted/470,490 chang
   ed from V2.0)
100 DIM L$(500),LNUM(500):COLO
    R 0,7,7:KEY OFF:CLS:MAX=0:
    LNUM(0)=65536!
110 ON ERROR GOTO 120:KEY 15,C
    HR$(4)+CHR$(70):ON KEY(15)
     GOSUB 640:KEY (15) ON:GOT
    O 130
120 RESUME 130
130 DEF SEG=&H40:W=PEEK(&H4A)
140 ON ERROR GOTO 650:PRINT:PR
    INT"Proofreader Ready."
150 LINE INPUT L$:Y=CSRLIN-INT
    (LEN(L$)/W)-1:LOCATE Y,1
160 DEF SEG=0:POKE 1050,30:POK
    E 1052,34:POKE 1054,0:POKE
     1055,79:POKE 1056,13:POKE
     1057,28:LINE INPUT L$:DEF
     SEG:IF L$="" THEN 150
170 IF LEFT$(L$,1)=" " THEN L$
    =MID$(L$,2):GOTO 170
```

```
1BØ IF VAL(LEFT$(L$,2))=Ø AND
    MID$(L$,3,1)=" " THEN L$=M
    ID$(L$,4)
2ØØ IF ASC(L$)>57 THEN 26Ø 'no
     line number, therefore co
    mmand
2Ø5 BL=INSTR(L$," "):IF BL=Ø T
    HEN BL$=L$:GOTO 2Ø6 ELSE B
    L$=LEFT$(L$,BL-1)
2Ø6 LNUM=VAL(BL$):TEXT$=MID$(L
    $,LEN(STR$(LNUM))+1)
21Ø IF TEXT$="" THEN GOSUB 54Ø
    :IF LNUM=LNUM(P) THEN GOSU
    B 56Ø:GOTD 15Ø ELSE 15Ø
22Ø CKSUM=Ø:FOR I=1 TO LEN(L$)
    :CKSUM=(CKSUM+ASC(MID$(L$,
    I))*I) AND 255:NEXT:LDCATE
     Y,1:PRINT CHR$(65+CKSUM/1
    6)+CHR$(65+(CKSUM AND 15))
    +" "+L$
23Ø GOSUB 54Ø:IF LNUM(P)=LNUM
    THEN L$(P)=TEXT$:GDTD 15Ø
    'replace line
24Ø GOSUB 58Ø:GOTO 15Ø 'insert
     the line
26Ø TEXT$="":FOR I=1 TO LEN(L$
    ):A=ASC(MID$(L$,I)):TEXT$=
    TEXT$+CHR$(A+32*(A>96 AND
    A<123)):NEXT
27Ø OELIMITER=INSTR(TEXT$," ")
    :COMMAND$=TEXT$:ARG$="":IF
     DELIMITER THEN COMMAND$=L
    EFT$(TEXT$,OELIMITER-1):AR
    G$=MIO$(TEXT$,DELIMITER+1)
     ELSE DELIMITER=INSTR(TEXT
    $,CHR$(34)):IF DELIMITER T
    HEN CDMMAND$=LEFT$(TEXT$,D
    ELIMITER-1):ARG$=MIO$(TEXT
    $,DELIMITER)
28Ø IF COMMANO$<>"LIST" THEN 4
    1Ø
29Ø OPEN "scrn:" FOR OUTPUT AS
     #1
3ØØ IF ARG$="" THEN FIRST=Ø:P=
    MAX-1:GDTO 34Ø
31Ø DELIMITER=INSTR(ARG$,"-"):
    IF OELIMITER=Ø THEN LNUM=V
    AL(ARG$):GOSU8 54Ø:FIRST=P
    :GDTO 34Ø
32Ø FIRST=VAL(LEFT$(ARG$,OELIM
    ITER)):LAST=VAL(MID$(ARG$,
    OELIMITER+1))
33Ø LNUM=FIRST:GOSUB 54Ø:FIRST
    =P:LNUM=LAST:GOSUB 54Ø:IF
    P=Ø THEN P=MAX-1
34Ø FOR X=FIRST TO P:N$=MID$(S
    TR$(LNUM(X)),2)+" "
35Ø IF CKFLAG=Ø THEN A$="":GOT
    D 37Ø
36Ø CKSUM=Ø:A$=N$+L$(X):FOR I=
    1 TD LEN(A$):CKSUM=(CKSUM+
    ASC(MIO$(A$,I))*I) AND 255
    :NEXT:A$=CHR$(65+CKSUM/16)
    +CHR$(65+(CKSUM AND 15))+"
     "
37Ø PRINT #1,A$+N$+L$(X)
3BØ IF INKEY$<>"" THEN X=P
39Ø NEXT :CLOSE #1:CKFLAG=Ø
4ØØ GOTO 13Ø
41Ø IF COMMANO$="LLIST" THEN O
    PEN "lpt1:" FOR OUTPUT AS
    #1:GOTO 3ØØ
42Ø IF COMMANO$="CHECK" THEN C
    KFLAG=1:GOTO 29Ø
43Ø IF COMMAND$<>"SAVE" THEN 4
    5Ø
44Ø GOSU8 6ØØ:OPEN ARG$ FOR OU
    TPUT AS #1:ARG$="":GOTO 3Ø
    Ø
45Ø IF COMMANO$<>"LOAO" THEN 4
    9Ø
46Ø GOSUB 6ØØ:OPEN ARG$ FOR IN
    PUT AS #1:MAX=Ø:P=Ø
47Ø WHILE NOT EOF(1):LINE INPU
    T #1,L$:BL=INSTR(L$," "):B
    L$=LEFT$(L$,BL-1):LNUM(P)=
    VAL(BL$):L$(P)=MID$(L$,LEN
    (STR$(VAL(BL$)))+1):P=P+1:
    WEND
4BØ MAX=P:CLDSE #1:GOTO 13Ø
49Ø IF COMMANO$="NEW" THEN INP
    UT "Erase program - Are yo
    u sure";L$:IF LEFT$(L$,1)=
    "y" OR LEFT$(L$,1)="Y" THE
    N MAX=Ø:LNUM(Ø)=65536':GOT
    O 13Ø:ELSE 13Ø
5ØØ IF COMMANO$="BASIC" THEN C
    DLOR 7,Ø,Ø:ON ERROR GOTO Ø
    :CLS:END
51Ø IF COMMAND$<>"FILES" THEN
    52Ø
515 IF ARG$="" THEN ARG$="A:"
    ELSE SEL=1:GOSU8 6ØØ
517 FILES ARG$:GOTO 13Ø
52Ø PRINT"Syntax error":GOTO 1
    3Ø
54Ø P=Ø:WHILE LNUM>LNUM(P) AND
     P<MAX:P=P+1:WEND:RETURN
56Ø MAX=MAX-1:FOR X=P TO MAX:L
    NUM(X)=LNUM(X+1):L$(X)=L$(
    X+1):NEXT:RETURN
5BØ MAX=MAX+1:FOR X=MAX TO P+1
     STEP -1:LNUM(X)=LNUM(X-1)
    :L$(X)=L$(X-1):NEXT:L$(P)=
    TEXT$:LNUM(P)=LNUM:RETURN
6ØØ IF LEFT$(ARG$,1)<>CHR$(34)
     THEN 52Ø ELSE ARG$=MIO$(A
    RG$,2)
61Ø IF RIGHT$(ARG$,1)=CHR$(34)
     THEN ARG$=LEFT$(ARG$,LEN(
    ARG$)-1)
62Ø IF SEL=Ø AND INSTR(ARG$,".
    ")=Ø THEN ARG$=ARG$+".8AS"
63Ø SEL=Ø:RETURN
64Ø CLOSE #1:CKFLAG=Ø:PRINT"St
    opped.":RETURN 15Ø
65Ø PRINT "Error #";ERR:RESUME
     15Ø
```

Program 3: Commodore Proofreader

By Phillip Nelson, Assistant Editor

```
1Ø VEC=PEEK(772)+256*PEEK(773)
   :LO=43:HI=44
2Ø PRINT "AUTOMATIC PROOFREADE
   R FOR ";:IF VEC=42364 THEN
   {SPACE}PRINT "C-64"
3Ø IF VEC=5Ø556 THEN PRINT "VI
   C-2Ø"
4Ø IF VEC=35158 THEN GRAPHIC C
   LR:PRINT "PLUS/4 & 16"
5Ø IF VEC=17165 THEN LO=45:HI=
   46:GRAPHIC CLR:PRINT"128"
6Ø SA=(PEEK(LO)+256*PEEK(HI))+
   6:ADR=SA
7Ø FOR J=Ø TO 166:READ BYT:POK
   E ADR,BYT:ADR=ADR+1:CHK=CHK
   +BYT:NEXT
8Ø IF CHK<>2Ø57Ø THEN PRINT "*
    ERROR* CHECK TYPING IN DATA
    STATEMENTS":END
9Ø FOR J=1 TO 5:READ RF,LF,HF:
   RS=SA+RF:H8=INT(RS/256):L8=
   RS-(256*HB)
1ØØ CHK=CHK+RF+LF+HF:POKE SA+L
   F,LB:POKE SA+HF,HB:NEXT
11Ø IF CHK<>22Ø54 THEN PRINT "
   *ERROR* RELOAD PROGRAM AND
   {SPACE}CHECK FINAL LINE":EN
   D
12Ø POKE SA+149,PEEK(772):POKE
    SA+15Ø,PEEK(773)
13Ø IF VEC=17165 THEN POKE SA+
   14,22:POKE SA+18,23:POKESA+
   29,224:POKESA+139,224
14Ø PRINT CHR$(147);CHR$(17);"
   PROOFREADER ACTIVE":SYS SA
15Ø POKE HI,PEEK(HI)+1:POKE (P
   EEK(LO)+256*PEEK(HI))-1,Ø:N
   EW
16Ø DATA 12Ø,169,73,141,4,3,16
   9,3,141,5,3
17Ø DATA 88,96,165,2Ø,133,167,
   165,21,133,168,169
18Ø DATA Ø,141,Ø,255,162,31,18
   1,199,157,227,3
19Ø DATA 2Ø2,16,248,169,19,32,
   21Ø,255,169,18,32
2ØØ DATA 21Ø,255,16Ø,Ø,132,18Ø
   ,132,176,136,23Ø,18Ø
21Ø DATA 2ØØ,185,Ø,2,24Ø,46,2Ø
   1,34,2Ø8,8,72
22Ø DATA 165,176,73,255,133,17
   6,1Ø4,72,2Ø1,32,2Ø8
23Ø DATA 7,165,176,2Ø8,3,1Ø4,2
   Ø8,226,1Ø4,166,18Ø
24Ø DATA 24,165,167,121,Ø,2,13
   3,167,165,168,1Ø5
25Ø DATA Ø,133,168,2Ø2,2Ø8,239
   ,24Ø,2Ø2,165,167,69
26Ø DATA 168,72,41,15,168,185,
   211,3,32,21Ø,255
27Ø DATA 1Ø4,74,74,74,74,168,1
   85,211,3,32,21Ø
28Ø DATA 255,162,31,189,227,3,
   149,199,2Ø2,16,248
29Ø DATA 169,146,32,21Ø,255,76
   ,86,137,65,66,67
3ØØ DATA 68,69,7Ø,71,72,74,75,
   77,8Ø,81,82,83,88
31Ø DATA 13,2,7,167,31,32,151,
   116,117,151,128,129,167,136
   ,137
```

Program 4: Apple Proofreader

By Tim Victor, Editorial Programmer

```
1Ø C = Ø: FOR I = 768 TO 768 +
   68: READ A:C = C + A: POKE I
   ,A: NEXT
2Ø IF C < > 7258 THEN PRINT "ER
   ROR IN PROOFREAOER OATA STAT
   EMENTS": ENO
3Ø IF PEEK (19Ø * 256) < > 76 T
   HEN POKE 56,Ø: POKE 57,3: CA
   LL 1ØØ2: GOTO 5Ø
4Ø PRINT CHR$ (4);"IN#A$3ØØ"
5Ø POKE 34,Ø: HOME : POKE 34,1:
    VTA8 2: PRINT "PROOFREAOER
   INSTALLEO"
6Ø NEW
1ØØ OATA 216,32,27,253,2Ø1,141
11Ø OATA 2Ø8,6Ø,13B,72,169,Ø
12Ø OATA 72,189,255,1,2Ø1,16Ø
13Ø OATA 24Ø,8,1Ø4,1Ø,125,255
14Ø OATA 1,1Ø5,Ø,72,2Ø2,2Ø8
15Ø OATA 238,1Ø4,17Ø,41,15,9
16Ø DATA 48,2Ø1,58,144,2,233
17Ø OATA 57,141,1,4,138,74
18Ø OATA 74,74,74,41,15,9
19Ø OATA 48,2Ø1,58,144,2,233
2ØØ DATA 57,141,Ø,4,1Ø4,17Ø
21Ø OATA 169,141,96
```

©

It Talks!
It Recognizes!
It Writes Music!
and more . . .

THE AMAZING VOICE MASTER®
Speech and Music Processor

Your computer can talk in your own voice. Not a synthesizer but a true digitizer that **records** your natural voice quality—and in any language or accent. Words and phrases can be expanded without limit from disk.

And it will understand what you say. A real word recognizer for groups of 32 words or phrases with unlimited expansion from disk memory. **Now you can have a two way conversation with your computer!**

Easy for the beginning programmer with new BASIC commands. Machine language programs and memory locations for the more experienced software author.

Exciting Music Bonus lets you hum or whistle to write and perform. Notes literally scroll by **as you hum!** Your composition can be edited, saved, and printed out. You don't have to know one note from another in order to write and compose!

Based upon new technologies invented by **COVOX. One** low price buys you the **complete system**—even a voice controlled black-jack game! In addition, you will receive a subscription to COVOX NEWS, a periodic newsletter about speech technology, applications, new products, up-dates, and user contributions. **You will never find a better value for your computer.**

ONLY $89.95 Includes all hardware and software.

For telephone demonstration or additional information, call **(503) 342-1271**. FREE audio demo tape and brochure available.

Available from your dealer or by mail. When ordering by mail add $4.00 shipping and handling ($10.00 for foreign, $6.00 Canada)

The Voice Master is available for the C64, C128, all Apple II's, and Atari 800, 800XL and 130XE. Specify model when ordering.

VISA MasterCard *For Faster Service on Credit Card Orders only:*

ORDER TOLL FREE 1-800-523-9230

COVOX INC. (503) 342-1271
675-D Conger Street, Eugene, OR 97402
Telex 706017 (AV ALARM UD)

Advertisers Index

Reader Service Number/Advertiser	Page
102 Abacus Software	37
103 Abacus Software	39
104 Artificial Intelligence Research Group	71
105 Atari	24–25
C O M 8 Direct Marketing Corp	67
Commodore	BC
106 CompuServe	1
ComputAbility	61
The Computer Book Club	27
Computer Direct	54–57
107 Computer Mail Order	14–15
Covox Inc	128
108 Duplicating Technologies, Inc	18
109 Electronic Arts	IFC
110 Electronic Arts	2–3
111 Elek-Tek, Inc.	51
112 EPYX	11
Halix Institute	118
113 J&R Music World	29
114 Kyan Software	53
Lyco Computer	62–63
115 Merdyne Publishers, Inc	66
116 99/4A National Assistance Group	118
NRI Schools	35
117 Precision Data Products	88
118 Professor Jones	128
119 Puma	19
120 Quinsept, Inc	59
121 subLOGIC Corporation	IBC
122 Timeworks, Inc	7
123 Unitech	118
124 White House Computer	106

COMPUTE! Classifieds	120
COMPUTE! Disk Subscription	17
COMPUTE! Subscription	33
COMPUTE!'s Apple Applications Special	49
COMPUTE!'s Elementary Amiga BASIC and Elementary ST BASIC	4
COMPUTE!'s IBM Books Collection	23
COMPUTE!'s Kids and the Atari ST and Kids and the 128	64
COMPUTE!'s Programmer's Guide	43
40 Great Flight Simulator Adventures	9
128 Machine Language for Beginners	13

PROFESSIONAL
HANDICAPPING SYSTEMS
• PRESENTED BY PROFESSOR JONES •

GLD. Thoroughbred "Gold" Edition™
A "Full" featured thoroughbred analysis designed for the professional and the serious novice
$159.95 complete

EGLD. Enhanced "Gold" Edition™
"Gold" Edition with complete Master Bettor™ system integrated onto the same disk. This powerful program will transfer all horses and scores to the bet analysis with a "single keystroke."
(Master Bettor™ included) **$199.95** complete

GLTD. Limited "Gold"™
Enables **Professional Handicappers** to assign specific values to the racing variables *they* feel are important. Create program weight based on a particular track and fine tune it for **maximum** win percentage. This program is designed for "ease of use." The user needs **no programming experience**.
(contains Integrated Bettor™) **$299.95** complete

GD. Gold Dog Analysis™ **$149.95**
The *ONLY* professional greyhound analysis available that evaluates *ALL* variables
w/integrated Bettor **$199.95**
Limited Version **$299.95**

MHH. Master Harness Handicapper™
Professional software designed to provide a thorough analysis of all trotter and pacer races in the United States and Canada
complete **$159.95**
w/integrated Bettor **$199.95**
Limited **$299.95**

Professor Pix Football™
Complete STATISTICAL ANALYSIS on Data Base allowing "Designated" previous games to be evaluated
Statistical Series **$99.95**
w/Win Loss Power Ratings **$149.95**

NBA. Basketball™ NBA **$99.95** w/college **$129.95**
w/power ratings **$149.95**

LOT. Lottery Analysis™
Statistical comparison program designed to detect subtle patterns in winning numbers and digits
Lottery (3-4 digit) **$79.95**
w/Lotto (Max 99 Digit) **$99.95**

PC-3 Portable Computer (4k) w/choice of Thoroughbred, Greyhound or Trotter™ **$249.95** (includes portable computer & program.)

M-100 Portable (32k) w/choice of Thoroughbred, Greyhound or Trotter and Master Bettor
$649.95
(includes portable computer and program.)

FREE CATALOG!

★ ★ ★ ★ ★
"Your BEST BET in handicapping software..."

Reviewed by:
Whole Earth Software Guide
Corona Data Systems

Prof. Jones
1940 W. State
Boise, ID 83702

VISA MasterCard
AMEX

48 HR. FREE SHIPPING

☎ CALL
(208) 342-6939

TERMS Free shipping all software. Add $6.00 hardware/$6.00 C.O.D./UPS Blue $6.00/Out of Country $9.00/ID Residents 4%/3 weeks personal checks/Cash price only, add 2½ Visa, MC, AMEX. Prices subject to change

Flight Simulator II Scenery Disks

The Challenge of Accomplished Flight

With a realism comparable to (and in some ways even surpassing) $100,000 aircraft flight simulators, Flight Simulator II includes full flight instrumentation and avionics, and provides a full-color out-the-window view. Instruments are arranged in the format standard to modern aircraft. All the radios needed for IFR flight are included. Front, rear, left, right, and diagonal views let you look in any direction. Program features are clearly documented in a 96-page Pilot's Operating Handbook.

For training in proper flight techniques, Flight Simulator II includes another 96-page instruction manual, compiled by two professional flight instructors with over 8,000 hours flight time and 12,000 hours of aviation teaching experience. You'll learn correct FAA-recommended flight procedures, from basic aircraft control through instrument approaches. To reward your accomplishments, the manual even includes a section on aerobatic maneuvers.

The Realism and Beauty of Flight

Go sight-seeing over detailed, realistic United States scenery. High-speed graphic drivers provide an animated out-the-window view in either day, dusk, or night flying modes.

Flight Simulator II features over 80 airports in four different scenery areas: New York, Chicago, Seattle, and Los Angeles. Six additional Scenery Disks covering the entire Western half of the United States are now available in IBM and C64/128 disk formats. Apple and Atari versions will be released soon. Each disk covers a geographical region of the country in detail, and is very reasonably priced.

The Pure Fun of "World War I Ace"

When you think you're ready, you can test your flying skills with the "World War I Ace" aerial battle game. This game sends you on a bombing run over heavily-defended enemy territory. Six enemy fighters will attempt to engage you in combat as soon as war is declared. Your aircraft can carry five bombs, and your machine guns are loaded with 100 rounds of ammunition.

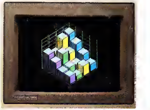
See Your Dealer. Flight Simulator II is available on disk for the Apple II, Atari XL/XE, and Commodore 64/128 computers for $49.95. Scenery Disks for the C64 and IBM PC (Jet or Microsoft Flight Simulator) are $19.95 each. A complete Western U.S. Scenery six-disk set is also available for $99.95. For additional product or ordering information, call (800) 637-4983.

Apple II is a trademark of Apple Computer, Inc
Atari XL and XE are trademarks of Atari Corp
Commodore 64 and 128 are trademarks of Commodore Electronics Ltd
IBM PC is a registered trademark of International Business Machines Corp

subLOGIC
Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex 206995

Order Line: (800) 637-4983
(except in Illinois, Alaska and Hawaii)

All you need to do this



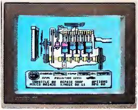
graph a spreadsheet

WELCOME TO JANEWRITE

This is an example of JANEWRITE the wordprocessor in JANE from Arktronics. This product will become a standard against which all picture based integrated software products will be measured.

Jane allows complete left and right margin control as well as the ability to set paragraph margins. The ability to simulate three different character sizes allows the user to have either a 40, 64, or 80 column screen. The product has a

write a novel



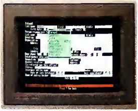
fix an engine

compose a song

paint a picture

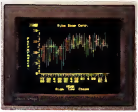
your banking

learn to fly



organize a data base

tell a story



forecast sales

is this.

When it comes to personal computers, you want the smartest, at a price that makes sense.

The new Commodore 128™ system has a powerful 128K memory, expandable by 512K. An 80-column display and 64, 128 and CP/M® modes for easy access to thousands of educational, business and home programs. And a keyboard, with built-in numeric keypad, that operates with little effort.

Or if the Commodore 128 is more machine than you had in mind, you can pick up the Commodore 64.® The Commodore 64 is our lower-priced model geared to more fundamental, basic needs.

Discover personal computers that do more for you. At prices you've been waiting for. From the company that sells more personal computers than IBM® or Apple.®

COMMODORE 128 AND 64 PERSONAL COMPUTERS

A Higher Intelligence

© 1985 Commodore Electronics Limited
® CP/M is a registered trademark of Digital Research Inc.
® Apple is a registered trademark of Apple Computer Inc.
® IBM is a registered trademark of
International Business Machines Corporation
® Commodore 64 is a registered trademark of Commodore Electronics Ltd.

RETROMAGS

Our goal is to preserve classic video game magazines so that they are not lost permanently.

People interested in helping out in any capacity, please visit us at www.retromags.com.

No profit is made from these scans, nor do we offer anything available from the publishers themselves.

If you come across anyone selling releases from this site, please do not support them and do let us know.

Thank you!